Universiteit van Amsterdam

Software Engineering MSc

Experimenteren met datamining technieken in een benchmark- en leer- systeem voor het Radiotherapeutisch netwerk

Auteur:
ing. Ray Burgemeestre

Opdrachtgever:
GreCom Application Development BV
in samenwerking met: MAASTRO clinic

Stagebegeleiding:
prof. dr. A. Hasman (AMC),
drs. H. Dekkers (UvA),
Jørgen van den Bogaard MA (MAASTRO),
ing. J. Hofstede (GreCom)

9 september 2008

Universiteit van Amsterdam (UvA), Academisch Medisch Centrum (AMC),
Klinische Informatiekunde (KiK), Maastricht Radiation Oncology (MAASTRO)

Voorwoord

Het voorwoord bewaar ik voor het allerlaatste moment!

Samenvatting

MAASTRO clinic is een radiotherapieinstelling die sinds 2005 op een systematische wijze (bijna)incidenten analyseert. De analyse vindt plaats volgens de PRISMA methode. Hierbij wordt gekeken wat de basisoorzaken zijn van een incident en wat de waarde van de contextvariabelen is. Doordat de basisoorzaken en contextvariabelen zijn geclassificeerd is het mogelijk om incidenten met elkaar te vergelijken en kernoorzaken van problemen te achterhalen. De resultaten van de analyse worden gebruikt om het proces te verbeteren.

Op dit moment wordt de analyse nog zeer beperkt ondersteund door automatische tools. Hierdoor is het mogelijk dat bepaalde patronen (een combinatie van basisoorzaken cq. contextvariabelen) die structureel leiden tot problemen niet onderkend worden. Verder vereist de huidige analyse veel vakmanschap; worden niet alle gegevensbronnen benut voor analyse; schaalt de analyse lastig op naar grotere gegevensbestanden; is niet altijd duidelijk of er sprake is van structurele dan wel incidentele problemen en is het vergelijken van gegevens met andere instellingen lastig.

In dit onderzoeksproject is gekeken naar technieken die kunnen ondersteunen bij de analyse. Hiertoe is een prototype ontwikkeld met de volgende features:

- Contextvariabelen worden inzichtelijk gemaakt middels tabellen en analyse is mogelijk door “drill-down” functionaliteit.
- Control Charts kunnen toegepast worden om te kijken of het proces onder controle is.
- N-Grams kunnen gebruikt worden om ook gedeeltelijke combinaties van basisoorzaken en contextvariabelen te herkennen.
- K-Means clustering kan gebruikt worden om nog minder evidente verbanden te herkennen.
- Text-mining maakt het mogelijk concept-analyse uit te voeren op kwalitatief slechte teksten, privacyproblemen bestaan niet in alle representaties van de concepten.

De analyse maakt het mogelijk om:

- kwaliteit van verschillende afdelingen met elkaar te vergelijken
- inzicht te krijgen in de ontwikkeling van de kwaliteit van een afdeling over de tijd
- de kwaliteit van verschillende radiotherapie instellingen met elkaar te vergelijken

De bruikbaarheid van het prototype is getest door analisten van de MAASTRO. De resultaten waren:

- De analist kan dankzij het prototype sneller analyseren en heeft meer analysemogelijkheden, zoals zoeken naar combinaties.
- Dankzij het prototype zijn in de gegevens nieuwe observaties gedaan die door de analist interessant geacht werden.
- De effecten van verbetermaatregelen kan de analist bestuderen in een Control Chart.
- K-Means clustering is niet bruikbaar wegens de slechte kwaliteit van de contextvariabelen.

Inhoudsopgave

1 Inleiding

Oorsprong van het onderzoek Dhr. Rein Willems (president-directeur Shell Nederland) vermeldt [41] dat Shell in 15 jaar een reductie van 75% in aantallen incidenten heeft weten te bewerkstelligen. Deze lijn van verbetering wordt in dit rapport mogelijk geacht voor de Zorg en zou een kostenbesparing van €1-3 miljard met zich meebrengen. Het ministerie van Volksgezondheid, Welzijn en Sport (VWS) heeft Dhr. Rein Willems binnen het programma 'Sneller Beter'[1] gevraagd om een advies met betrekking tot patiëntveiligheid.

Het rapport formuleert een viertal adviezen, waarvan één aandringt op het per 1 januari 2008 verplicht stellen van de invoering van een Veiligheidsmanagement systeem (VMS). Aanleiding tot dit advies is de behoefte vanuit de medische gemeenschap om na het registreren en analyseren van (bijna-) incidenten (onderdeel van een VMS), hier ook zoveel mogelijk van te kunnen leren. Dit is te zien in initiatieven zoals het platform patiëntveiligheid [2] en het gezamenlijke patiëntveiligheidstraject in de radiotherapie [25]. Leren op het niveau van de zorginstelling tussen de afdelingen middels benchmarking alsmede de mogelijkheid dit op landelijk niveau te kunnen doen met andere zorginstellingen.

Aanleiding tot het onderzoek Het management binnen GreCom wilt onderzoeken in hoeverre een "online benchmarking platform"-systeem kan helpen bij het leren van ieders bevindingen, en bij het de kwaliteitsverbeteringtrajecten die men zal starten.

Opbouw van deze scriptie Elk hoofdstuk begint met een korte samenvatting van de inhoud, waar dieper op ingegaan wordt in de secties en subsecties. In hoofdstuk 2 worden definities gegeven van begrippen die in de rest van de scriptie worden gehanteerd. Daar staat tevens achtergrondinformatie in met betrekking tot o.a., de PRISMA systematiek. In hoofdstuk 3 wordt ingegaan op het doel van het onderzoek, de onderzoeksvraag, subonderzoeksvragen en onderzoeksmethode. De volgende drie hoofdstukken, 4, 5 en 6, gaan in op de drie subonderzoeksvragen. De resultaten volgen in hoofdstuk 7, daarna een evaluatie en conclusie in hoofdstuk 8. ■

[1] Website "Sneller Beter": `http://www.snellerbeter.nl`.
[2] Website "Platform patiëntveiligheid": `http://www.platformpatientveiligheid.nl`.

2 Achtergrond en Context

2.1. Patiëntveiligheid

In het "Sneller beter" rapport [1] is uitgegaan van de schatting dat er in Nederland tussen de 1.500 en 6.000 patiënten overlijden als gevolg van incidenten die voorkomen hadden kunnen worden. In april 2007 is de schatting bijgesteld op basis van het onderzoek "Onbedoelde schade in Nederlandse ziekenhuizen" [18] van het NIVEL en het VUmc in opdracht van de Orde van Medisch Specialisten [2]. Naar schatting overlijden 1737 patiënten als gevolg van een vermijdbare complicatie. Zo'n 76.000 opgenomen patiënten lopen onbedoeld schade op, van 30.000 had dit vermeden kunnen worden, bij 10.000 is de schade blijvend. Deze vermijdbare complicaties kosten de zorg 167 miljoen euro.

De hier gegeven definities komen uit het *Praktijkboek patiëntveiligheid* [34, hoofdstuk 1 & 6]. Een aantal relaties tussen deze definities zijn in figuur 2.1 weergegeven (de verhoudingen zijn niet reeël). In deze scriptie wordt voornamelijk over incidenten gesproken en ik wil benadrukken dat dit voornamelijk om "near misses" gaat waar geen sprake is van schade.

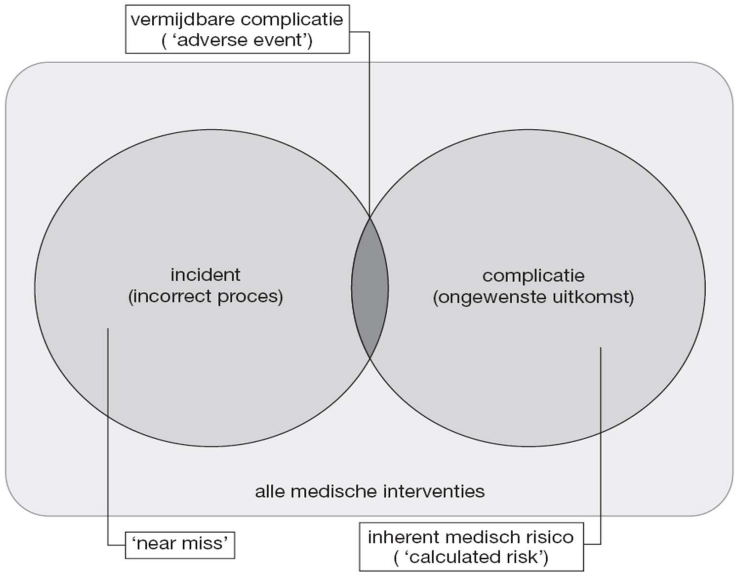
Patiëntveiligheid (patient safety) het (nagenoeg) ontbreken van (de kans op) aan de patiënt toegebrachte schade (lichamelijk/psychisch) die is ontstaan door het niet volgens de professionele standaard handelen van hulpverleners en/of door tekortkoming van het zorgsysteem.

Professionele standaard de beste manier van handelen in een specifieke situatie met inachtneming van recente inzichten en evidence zoals neergelegd in richtlijnen en protocollen van de beroepsgroep, dan wel het handelen zoals van een redelijk ervaren en bekwame beroepsgenoot in gelijke omstandigheden mag worden verwacht.

Veiligheidsmanagementsysteem (VMS) een ∼ is een bedrijfsfunctie die gericht is op het structureel en planmatig in kaart brengen, analyseren, verbeteren en borgen van de veiligheid van patiënten en medewerkers. In dat systeem worden risico's preventief geïnventariseerd, incidenten en interventies continu of periodiek in kaart gebracht en geanalyseerd en knelpunten via systematische verbetering aangepakt; de bereikte resultaten worden breed ingevoerd en opgenomen in het borgingssysteem.

Adverse event een onbedoelde gebeurtenis (procesgang) die is ontstaan door het (niet) handelen van een zorgverlener en/of door het zorgsysteem met schade (uitkomst) voor de patiënt, zodanig ernstig dat er sprake is van tijdelijke of permanente beperking, verlenging of verzwaring van de behandeling dan wel overlijden van de patiënt.

[1] Website "Sneller Beter": `http://www.snellerbeter.nl`.
[2] Zie ook het volgende nieuwsbericht: `http://www.bijzijn.nl/content/html/237.asp?nb_id=7165`



Figuur 2.1: Globale relatie tussen de beide hoofdconcepten incidenten en complicaties en de deelconcepten 'near miss', 'calculated risk' en 'adverse event' (bron: [34, p. 10]).

Complicatie een onbedoelde en ongewenste uitkomst tijdens of volgend op het handelen van een zorgverlener, die voor de gezondheid van de patiënt zodanig nadelig is dat aanpassing van het (be)handelen noodzakelijk is dan wel dat sprake is van onherstelbare schade.

Incident (event) een onbedoelde gebeurtenis tijdens het zorgproces die tot schade aan de patiënt heeft geleid, had kunnen leiden of (nog) kan leiden.

Schade (injury) een nadeel voor de patiënt dat door zijn ernst leidt tot verlenging of verzwaring van de behandeling, tijdelijk of blijvend lichamelijk, psychisch en/of sociaal functieverlies, of tot overlijden.

Procesafwijking afwijking van het geplande, verwachte of vereiste proces door (niet) handelen van een hulpverlener.

Vermijdbaar (preventable) een incident, complicatie of adverse event is in retrospectie vermijdbaar als na systematische analyse van de gebeurtenis(-sen) blijkt dat bepaalde maatregelen het incident, de complicatie of de adverse event hadden kunnen voorkomen.

2.2. Het benchmarkpunt

Definities 'benchmark' en 'benchmarking' Definities op internetpagina's en in verschillende wetenschappelijke boeken (via Google books) zijn in de kern gelijk aan de definitie in de Van Dale:

benchmark (het ˜):1. norm waarmee prestaties van producties worden vergeleken.
benchmarking (de ˜):1. vergelijking met bedrijven uit dezelfde sector.

De nadruk wordt vaak gelegd op het feit dat een benchmark een meting is, 'measurements to gauge the performance of a function, operation, or business relative to others'. Wat ook vaak benadrukt wordt is dat benchmarking een (doorlopend) proces is t.b.v. continue verbetering. Op verschillende typen benchmarking wordt kort ingegaan in hoofdstuk 6, sectie 6.1.

In alle gevoerde gesprekken in het vooronderzoek is het concept "landelijk benchmarkpunt" niet eenduidig gedefinieerd. Veel uiteenlopende interpretaties bestonden naast elkaar. Hier volgt een vereniging in een definitie die in deze scriptie gehanteerd zal worden.

Definitie "landelijk online benchmarkpunt" Een voor systeem dat beschikbaar is voor verpleegkundigen, artsen en medisch specialisten en dat verschillende zorginstellingen met elkaar in contact kan brengen. Het doel van het benchmarkpunt is dat de gebruiker kan leren van andere afdelingen en/of zorginstellingen op een veilige en geanonimiseerde manier. Het gaat om het leren van initiële incidenten registraties, de daaropvolgende: analyses, rapportages, ingeslagen verbetertrajecten, evaluaties, *et.al.* Dit gebeurd op basis van vergelijkingen gemaakt door het systeem en het delen van informatie in een online community. Daarnaast kan het benchmarkpunt twee anonieme partijen met elkaar in contact brengen zodat buiten het systeem om dialogen gestart kunnen worden.

In een benchmarkpunt kan op verschillende manieren geleerd worden. Ten eerste aan de hand van het door het systeem herkende trends in incidenten registraties, zo kan beter worden beoordeeld of problemen structureel zijn en een aanpassing van het proces nodig is dan wel van incidentele aard zijn. Ten tweede uit reeds door instituten geïdentificeerde faalwijzen en oorzaken alsmede de ingeslagen verbetertrajecten en evaluaties. Ten derde uit actieve uitwisseling van informatie zoals rapportages, metrieken, *etc.* Tot slot uit de dialogen gestart met behulp van het systeem, dat statistisch gezien de meest ideale instituten heeft uitgekozen.

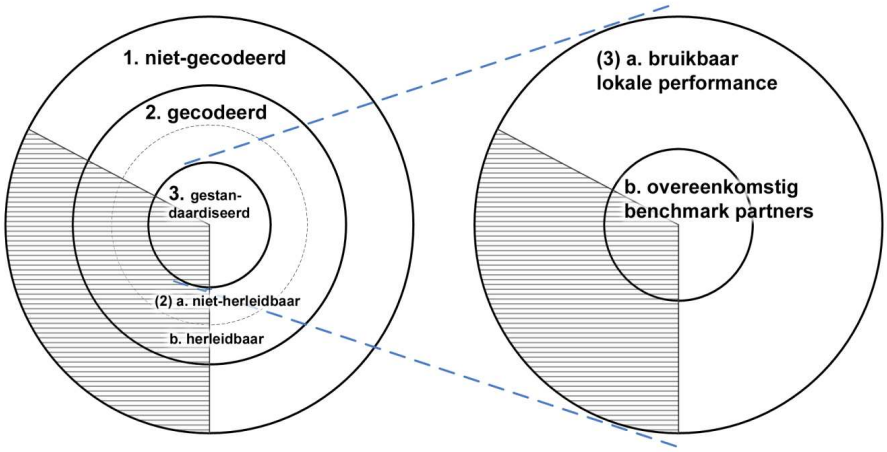
Hieruit kunnen voorzichtig een aantal zaken afgeleid worden:

- Hoe meer deelnemers, hoe meer potentie tot het daadwerkelijk komen tot leren van anderen.
- Hoe diverser de deelnemers, hoe algemener de vergelijkingen in het systeem moeten zijn als iedereen mee moet doen in de vergelijkingen.
- Hoe meer informatie gedeeld wordt in het systeem, hoe belangrijker het regelen van anonimiteit wordt.
- Het systeem kan faciliteren in het verlagen van de drempel tussen zorginstellingen om op basis van bevindingen in het systeem in contact te treden, op een manier die beide partijen vertrouwen.

Definitie gegevensbron Verzamelde gegevens binnen een VMS waarvan de oorsprong is de meldingen en alle mogelijke aanvullingen hierop. Verwijzingen naar informatie in andere systemen zelf vallen hier ook onder, de informatie waar naar verwezen wordt niet.

Visueel zijn weergegeven de verschillende elementen waaruit de gegevensbron kan bestaan in figuur 2.2. Dit is gedaan omdat het type gegevens sterk uiteen kan lopen, van vrije tekst tot strikte classificatiecodes. In het figuur wordt bijvoorbeeld bedoeld: plaintext vrije tekst (1.); meerkeuzevragen (2.) zijnde wel (2.a.) of niet (2.b.) herleidbaar tot een specifieke organisatie of gevoelige informatie (betrokkenen, ...); en "harder" gedefinieerde codes/classificaties (3.) die weer wel (3.a.) of niet (3.b.) bruikbaar zijn voor (externe) benchmarking. Of met andere woorden wel of niet overeenkomstig tussen de benchmarkende partijen.

Niet bruikbaar voor benchmarking betekend in dit geval dat voor de twee instituten kennis nodig hebben van een intern proces (*i.e.*, vergelijken van uitvoerende werkeenheid), dit is niet altijd het geval omdat elk instituut zijn eigen processen inricht. Tweede rede dat bepaalde gegevens niet bruikbaar zijn is dat dezelfde gegevens niet aanwezig zijn voor beide instituten.

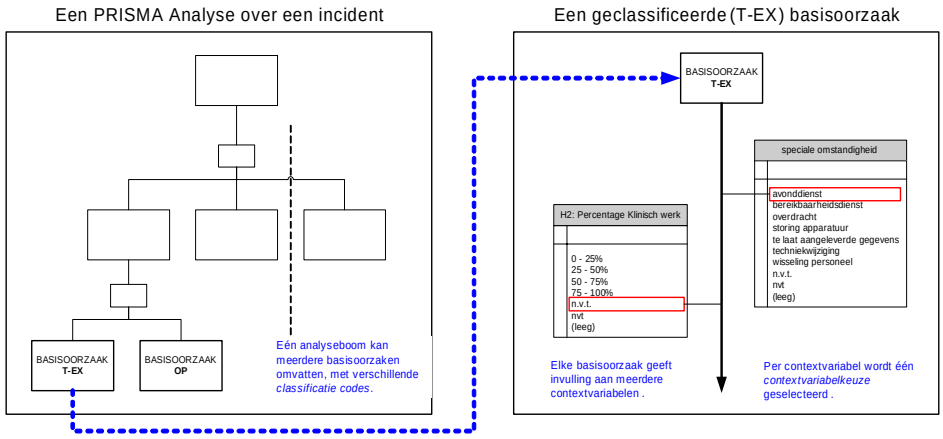


Figuur 2.2: Typen gegevensbronnen (de grijze gebieden geven de gedeelten aan die niet bruikbaar zijn).

2.3. PRISMA systematiek

De volgende tekst is een samenvatting van de PRISMA methodiek gebaseerd op [34, h. 21] en [33][3].

PRISMA structureert en rubriceert meldingen van incidenten. De afkorting staat voor Prevention and Recovery Information System for Monitoring and Analysis. Het brein achter PRISMA is prof.dr. T.W. van der Schaaf. Dit wordt gedaan door het opbouwen oorzakenboom (figuur 2.3) Deze wordt gekenmerkt door oorzaken en basisoorzaken (de leaves) en een herstelzijde. De herstelzijde is dat gedeelte waarin gemodelleerd wordt wat heeft doen voorkomen dat het mis ging of gedaan moet worden om te voorkomen dat het mis gaat.



Figuur 2.3: Conceptuele voorstelling relatie analyse, basisoorzaken en contextvariabelen.

De boom wordt d.m.v. root-cause analysis opgebouwd, doorgaan wordt met "waarom?" vragen totdat de twee *stopregels* bereikt worden (vrij vertaald uit [33]):

[3] Zie voor een uitgebreidere omschrijving ook deze referentie, hier wordt in zeven pagina's een "brief description" gegeven door van der Schaaf.

- Stop het uitbreiden van de boom indien er geen objectieve feiten meer naar voren gebracht kunnen worden.
- Stop met het zoeken naar oorzaken van oorzaken indien de systeem grens is gepasseerd, dat is wanneer de benodigde maatregelen buiten het bereik van invloed vallen van de organisatie.

Volgens de systematiek moeten de basisoorzaken geclassificeerd worden met behulp van het uitvoerig geteste Eindhoven Classificatie Model (zie appendix A). Herstelfactoren kunnen ook geclassificeerd worden op "planned" en "not-planned". Om de analist te helpen is hiervoor een beslisboom beschikbaar (zie ook appendix A).

Het PRISMA profiel (een specifiek histogram) kan opgebouwd worden, na het uitvoeren van een aantal analysen (minimaal vijftig). De classificaties komen over de X-as te staan, en de hoe vaak deze als classificatie gebruikt zijn op de Y-as, zie B.4 in appendix B.

Op basis van het PRISMA profiel kan gestuurd worden door aan te nemen dat de gewenste situatie een is waar *e.g.* alle classificaties even veel scoren (een uniforme distributie) de ideale situatie vertegenwoordigd. Waar hoger wordt gescoord kan aan de hand van een actie/interventie-matrix (opgenomen in appendix A) grof bepaald worden wat voor type interventie nodig is. Een voorbeeld: In het geval van een hoge T-EX (Technisch falen Extern) score wordt escalatie aanbevolen (probleem laten afhandelen op hoger managementniveau in de organisatie).

De interventie-matrix kan niet altijd letterlijk gevolgd worden, daarom wordt geadviseerd in [33] om extra contextinformatie bij te houden.

Basisoorzaak Een basisoorzaak is een oorzaak binnen de root-cause analysis boom, die volgens de *stopregels* niet nog meer is onder te verdelen in onderliggende oorzaken. Synoniem: faalwijze.

2.3.1. PRISMA in de radiotherapie

De MAASTRO heeft een set contextvariabelen opgesteld, die later door het ZRTI ook in gebruik is genomen. De bedoeling is dat voor *i.e.*, een classificatie deze contextinformatie verder uitsluitsel kan geven m.b.t. de oorzaak. In deze scriptie worden de volgende begrippen gebruikt, zie ook figuur 2.3.

Contextvariabel Een contextvariabel is een afgesproken "variabele" dat iets over de context kan vertellen van een basisoorzaak. Onder context wordt verstaan "wie?, wat?, waar?, hoe?, wanneer?, ...', en een variabele kan antwoord geven op *e.g.*: "welke werkeenheid?", "wat voor apparatuur?". Het antwoord op deze vragen wordt gegeven door voorgedefinieerde contextvariabelen.

In de PRISMA RT (Radiotherapie) zijn de gehanteerde contextvariabelen als volgt ingevuld. Concrete contextvariabelen zijn *e.g.*: "apparatuur jonger dan één jaar", "apparatuur ouder dan één jaar". Deze variabelen horen tot één groep, de "apparatuur leeftijd". Per groep werd altijd hoogstens één contextvariabel toegewezen aan een basisoorzaak.

Contextvariabelmodel Een afgesproken en statische set contextvariabelen dat beschikbaar is om te koppelen aan basisoorzaken. Een model kan niet halverwege gewijzigd worden, een wijziging betekend een nieuw(e) (versie van het) model. Een model kan intern (op het niveau van een instituut) anders zijn dan extern afgesproken. Strikt genomen kan alleen tussen twee dezelfde modellen gebenchmarkt worden. [4]

[4] De stuurgroep van het Radiotherapie netwerk is bezig met de ontwikkeling van een dynamischer model dat gebruikt moet gaan worden op het moment van de landelijke in gebruik neming van het benchmarkpunt prototype 1 januari 2009.

In de PRISMA RT is gekozen voor zo min mogelijk contextvariabelen, omdat een gedetailleerd model te veel tijd in kan nemen en het analyseproces misschien ingewikkelder maakt. Hoe gedetailleerder de beschikbare set van contextvariabelen, des te nauwkeuriger kan de context gegeven worden, echter hoe meer tijd het koppelen van de juiste contextvariabelen aan de basisoorzaken gedurende de analyse zal innemen.

2.3.2. Dataverzameling PRISMA gegevensbron

Artsen op een afdeling voeren behandelingen uit, hierbij zijn betrokken machines, mensen, processen, protocollen, *etc.* Als een behandeling leidt tot een incident, maakt een betrokkene een melding en later voert een PRISMA analist de root-cause analyse uit. In deze analyse wordt naar oorzaken gevraagd. Deze oorzaken worden geclassificeerd als basisoorzaken, naast een classificatie worden tegelijk ook contextvariabelen toegewezen.

Na synchronisatie komen deze basisoorzaken in een database terecht. Daar worden ze opgeslagen per melding (id) en instituut (id). In deze database zijn ook de contextvariabelen weer toegewezen aan de basisoorzaken. De omschrijvingen van deze basisoorzaken zijn ook opgenomen in de database, echter zijn de instellingen niet van plan deze teksten met elkaar te vergelijken. Ze zijn er voor het instituut zelf om dieper inzicht te krijgen in de geselecteerde classificaties door het lezen van de omschrijvingen. Van de classificaties kan een histogram gemaakt worden (classificaties over de x-as, aantallen over de y-as).

Deze visuele weergave wordt het PRISMA profiel (zie figuur B.4, appendix B) genoemd en staat centraal in de huidige werkwijze. Op het profiel wordt gestuurd (uitschietende classificaties terugdringen op basis van de "interventiematrix", appendix A), vergeleken (het profiel per kwartaal, of tussen instituten).

2.4. Overige definities

Definitie ontologie In de informatica is een ontologie het product van een poging een uitputtend en strikt conceptueel schema te formuleren over een bepaald domein. Een ontologie is typisch een hiërarchische datastructuur die alle relevante entiteiten en hun onderlinge relaties en regels binnen dat domein bevat, zoals bij een domein ontologie het geval is. Het gebruik van het woord binnen de informatica is afgeleid van het veel oudere gebruik van het woord ontologie binnen de filosofie. (bron: Wikipedia).

Voorbeelden van ontologieën:

- Taxonomieën op het web: Yahoo! categories
- Catalogi/ voor online shopping: Amazon.co.uk product catalog.
- Domeinspecifiek / standaard terminologie: SNOMED Clinical Terms: terminologie voor klinische medicatie of UNSPSC: terminologie voor producten en diensten. ■

(RD, dd. 7.feb. 2008)

3 Onderzoeksmethode

3.1. Literatuur- en vooronderzoek

Doel van het literatuuronderzoek was een verkenning van de onderwerpen waren[1]: patiëntveiligheid, benchmarking/statistiek, hoe anderen het doen met Performance Indicatoren, in de Process Improvement en middels Data Mining, social software, architectuur en Veiligheidsmanagement systemen. Een aantal thema's die onderkend zijn gedurende het literatuuronderzoek:

- *Zoeken en vinden van informatie.* De gegevensbronnen zijn uiteenlopend wat betreft de informatie dat geregistreerd wordt, maar ook het gebruik van contextinformatie kan uiteenlopen. Trefwoorden: ontologieën, taalanalyse.
- *Visualiseren van informatie.* Welke best-practices uit andere kennisgebieden zijn wel of niet bruikbaar? Verschillende visualisaties kunnen vergeleken worden en beoordeeld op basis van een aantal criteria. Trefwoorden: visualiseren.
- *Vergelijken van informatie.* Wat is de handigste methode om dit te realiseren en hoe kan een software systeem hieraan bijdragen? Trefwoorden: anonimiseren.
- *Leren in een anonieme omgeving.* Als alles anoniem moet zijn en de gegevens geanonimiseerd, gaat dan niet de belangrijke informatie verloren? Immers context informatie kan ook leiden tot ontmaskering van de zorginstelling. In het specifieke draagt het enorm bij aan het snappen van het probleem, in het algemene kan de informatie nog steeds onmisbaar zijn. Trefwoorden: anonimiseren/integreren.
- *Architectuur landelijk benchmarkplatform.* Een platform dat werkelijk landelijk vergelijking levert voor (naar mijn weten) ongeveer 180 ziekenhuizen met elk (informele schatting) honderden tot duizenden meldingen per jaar. Dat vervolgens allerlei wiskundige formules moet kunnen loslaten op de resulterende gegevensbron die alleen maar zal toenemen heeft een doordachte software architectuur nodig.

In overleg met GreCom, de MAASTRO, de UvA en het AMC is de focus uiteindelijk komen te liggen op "Vergelijken van informatie" en het "Visualiseren van informatie", binnen de PRISMA methode.

3.2. Doel van het onderzoek

De wens en het doel van de Radiotherapie-instellingen is patiëntveiligheid en dat wordt bewerkstelligd door het terugdringen van vermijdbare complicaties. De verwachting is dat benchmarking nog meer interessante inzichten kan opleveren die leiden tot het vergroten van de patiëntveiligheid (zoals in [25]). Het samenvoegen van data—op een verantwoorde manier—kan leiden tot nieuwe verbanden en hopelijk inzichten. Wie en waar binnen de radiotherapie beter presteert is zinvolle input voor een nader onderzoek.

[1] Literatuur dat uitsluitend in het vooronderzoek is geraadpleegd: process improvement: [19] en een aantal internetartikelen e.d., [23], [22], [32], [31]; social software: [10]; E-learning en E-Health [28], [26]; architectuur: C2 architectural style en middleware [13], [30], [14].

Het doel van dit onderzoek is de geschiktheid van de PRISMA RT gegevensbron toetsen met betrekking tot een landelijk benchmarkpunt. Vervolgens welke informatie uit deze gegevensbron te halen valt, onder andere met behulp van technieken die nog niet op dit soort type gegevens uitgeprobeerd zijn. Tot slot kijken welke informatie en op wat voor manier deze toegepast kan worden in benchmarking.

3.3. Probleemstelling

De onderzoeksvraag luidt: *"Hoe kunnen de PRISMA-analyse gegevens voor het Radiotherapeutisch netwerk gebruikt worden voor benchmarking op basis van data-analyse en data-mining?"*. De rede dat gekozen is voor PRISMA gegevens is de beschikbaarheid van de data en hier waren al ontwikkelingen in gang gezet (door de landelijke best-practice instellingen op het gebied van patiëntveiligheid binnen de radiotherapie).

In de onderzoeksvraag wordt onder data-*analyse* verstaan het drill-down proces als oorzakenanalyse, visualisaties, *etc.*, onder data-*mining* wordt verstaan het loslaten van algoritmen en formules op de data met als doel nog meer informatie te verkrijgen uit de gegevens, deze kunnen weer toegepast worden voor benchmarking.

Subonderzoeksvragen zijn:

Sub-1: Hoe zien de gegevens er *kwalitatief* en *kwantitatief* uit[2], welke zijn geschikt voor benchmarking en hoe kan voor de uitkomsten beoordeeld worden of deze voor het plegen van interventies (procesaanpassingen) efficiënt [3] zijn?

Sub-2: Welke informatie kan met behulp van data-mining technieken uit dezelfde gegevensbron gehaald worden, wat is daarvan geschikt voor benchmarking en hoe kunnen we valideren of de gevonden uitkomsten bruikbaar zijn voor het plegen van interventies (procesaanpassingen)?

Sub-3.1: Hoe kunnen we de *huidige werkwijze*[4] van de PRISMA analist automatiseren en ondersteunen (naar mate de selectie specifieker wordt neemt de schaarsheid van de metingen toe) in de data-analyse (drilldown deductieproces) en benchmarking, wat is de toegevoegde waarde [5]?

Sub-3.2: Hoe kunnen als aanvulling op de *huidige werkwijze*[6] de uitkomsten van Sub-1 (kansberekening) en Sub-2 (data-mining technieken) toegepast worden in het deductieproces en benchmarking, wat is de toegevoegde waarde[7]?

3.4. Onderzoeksmethode

In figuur 3.1 zijn de doorlopen fasen in het onderzoek visueel weergegeven. Bronnen van informatie zijn geweest wetenschappelijke literatuur (ACM, IEEE, PubMed) en een aantal boeken over onderwerpen patiëntveiligheid, statistiek, procesverbetering, *etc.*; een aantal brainstormsessies binnen GreCom, de MAASTRO en de UvA; een paar webconferences met de MAASTRO en tot slot ongeveer vijftig voortgangsbesprekingen (waarvan veertig uitgewerkt in gespreksverslagen) met mijn begeleiders vanuit het AMC prof.dr. Arie Hasman, vanuit de UvA drs. Hans Dekkers en vanuit GreCom ing. Julien Hofstede.

[2] zie hoofdstuk 4 voor een exactere toelichting.

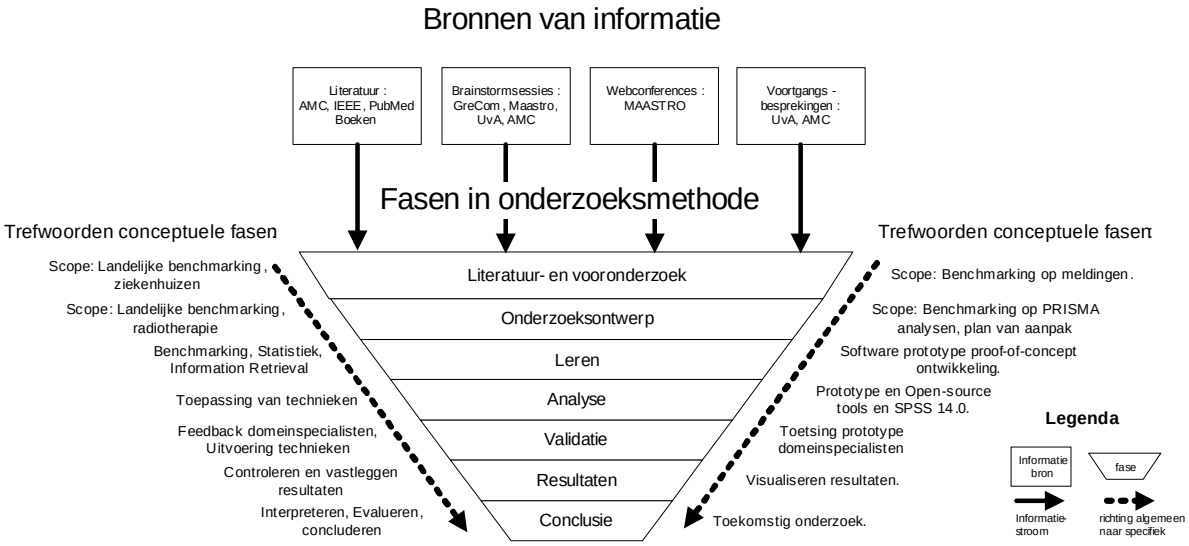
[3] De focus in dit onderzoek ligt vooral op efficiëntie en niet op effectiviteit. Om te bepalen of een interventie effectief is zijn indicatoren nodig zoals de ernstigheid van het incident. Deze informatie ligt niet eenvoudig voor handen vanuit een technisch oogpunt bezien. Dit en het feit dat we ook hoofdzakelijk te maken hebben met "bijna-incidenten"—en hier statistiek bedrijven op "basisoorzaken"—was het voor de hand liggender om te kiezen voor een focus op efficiëntie.

[4] *i.e.*, sectie 2.3.2

[5] zie definitie 3.4.1.

[6] *ibid., 4*

[7] *ibid., 5*



Figuur 3.1: Onderzoeksmethode trechter en informatie naast de conceptuele fasen voor de beeldvorming.

Het onderzoek is een exploratief en kwalitatief onderzoek. In de eerste fase zijn de richtingen onderkend (sectie 3.1). Hierna is het onderzoek specifieker gedefinieerd, *i.e.*, toegespitst op de Radiotherapie. Een fase van leren volgde omdat een aantal onderwerpen niet in de opleiding zijn behandeld (statistiek, patiëntveiligheid, data mining).

Hierop volgde analyse met betrekking tot de vorm van de data in de (PRISMA) gegevensbron, hier is bepaald wat precies voor benchmarking gebruikt kan worden en hoe deze op een verantwoorde en logische manier voor benchmarking gebruikt konden worden. Ook wordt hier omschreven hoe een aantal technieken uit onder andere datamining toegepast kunnen worden op de gegevens. Parallel zijn een aantal technieken direct in een prototype geïmplementeerd en een aantal opensource tools zijn gekoppeld aan de PRISMA gegevensbron.

In de validatiefase hebben de domeinexperts (MAASTRO clinic) de technieken op waarde getoetst middels het uitvoeren van gestructureerde analysen. Ook zijn met dezelfde domeinexperts webconferences gehouden op afstand (vanuit GreCom met de MAASTRO) en zijn visualisaties gegenereerd in SPSS alsmede de open-source tools.

De resultaten zijn genoteerd, geïnterpreteerd en gecontroleerd. Tot slot zijn deze geëvalueerd en zijn een aantal nieuwe waardevolle ideeën geformuleerd voor toekomstig onderzoek.

Op de drie subonderzoeksvragen Sub-1, Sub-2 en Sub-3 wordt in de corresponderen hoofdstukken, respectievelijk 4, 5 en 6, antwoord gegeven.

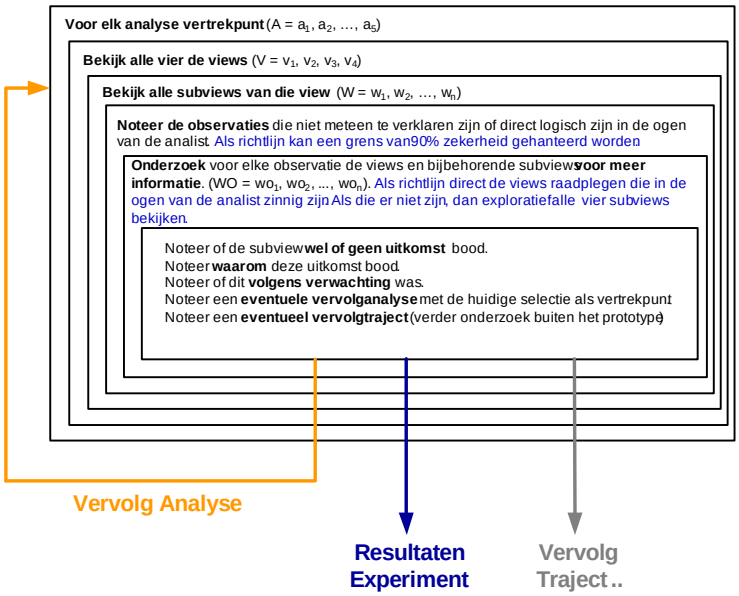
Evaluatie in het algemeen Evaluatie heeft op de volgende drie manieren plaatsgevonden. Ten eerste wordt elke behandelde techniek in het betreffende hoofdstuk zelf geëvalueerd en worden daar (sub)conclusies geformuleerd. Dit op basis van resultaten in SPSS of opensource tools. Ten tweede zijn een hoop zaken in één van de vele voortgangsbesprekingen geëvalueerd. Ten derde is het prototype geëvalueerd in samenwerking met de MAASTRO (zie volgende subsectie 3.4.1).

3.4.1. Evaluatie prototype

Het prototype is een software programma geschreven in PHP 5.0 dat gebouwd is op het framework van GreCom. Implementatiedetails zijn weggelaten in deze scriptie, een beknopte architectuurbeschrijving is opgenomen in appendix B. Het programma bevat hoofdzakelijk vier "views", die elk een aantal technieken implementeren.

Een experiment is opgesteld om de *toegevoegde waarde* van dit prototype te achterhalen voor de analist. De met het prototype uitgevoerde analysen hebben toegevoegde waarde als met het systeem "gevonden uitkomsten" ("opmerkelijke observaties") gevonden worden, die nog *niet* bekend waren en die *niet* door de analist ontdekt hadden kunnen worden zonder het systeem (vanwege tijd en moeite benodigd zonder systeem). In dit experiment staat het gebruik van een draaiboek (of script) centraal.

Een script wordt gebruikt dat ervoor zorgt dat functionaliteiten van het prototype niet over het hoofd gezien worden tijdens het analyseren. De analysen (en eventueel vervolganalysen) zullen op basis van dit script van zoveel mogelijk structuur worden voorzien, zie figuur 3.2. De uitvoering van analysen volgens dit draaiboek zullen in de resultaten worden besproken (hoofdstuk 7) en de directe uitkomsten zijn te vinden in appendix C.



Figuur 3.2: Experiment draaiboek gestructureerd analyseren.

4 Analyse PRISMA gegevensbron

Dit hoofdstuk beantwoord subonderzoeksvraag Sub-1 (opgedeeld):

- Hoe zien de gegevens er *kwalitatief* en *kwantitatief* uit?
- Hoe kan voor de uitkomsten van benchmarking beoordeeld worden of deze voor het plegen van *interventies* (procesaanpassingen) *efficiënt*[1] zijn?

De gegevens bestaan uit een paar duizend geclassificeerde basisoorzaken per jaar en volgen in veel gevallen een Poisson verdeling. Dit zijn alle classificaties uit het Eindhoven Classificatie Model zoals genoemd in hoofdstuk 2. De gegevens komen voort uit discrete metingen en hebben een statistisch karakter. Ze blijken na het kiezen van het interval van een week de Poisson verdeling te volgen.

Niet elke classificatie of groepering daarvan volgt een Poisson verdeling. Bijvoorbeeld de histogrammen van de menselijke classificaties Human/Intervention (HRI) en Human/Verification (HRV) voldoen niet aan de Kolmogorov-Smirnov test voor een Poisson verdeling, waar de overige menselijke classificaties dat wel doen. Wanneer we kijken naar alle 22 classificaties blijken 5 niet aan de test te voldoen.

Alles blijkt terug te brengen tot enkele classificaties die niet overeenkomen. Groepen van classificaties komen niet overeen omdat onderliggende classificaties dat niet doen. De niet-Poisson verdeelde menselijke classificaties HRI en HRV zorgen ervoor dat de groep van menselijke classificaties dat ook niet is. Daarom willen we deze classificaties kunnen wegfilteren.

De classificaties die niet aan de Poisson verdeling voldoen worden weggefilterd door middel van de Kolmogorov-Smirnov goodness-of-fit test zodat niet voortboorduurd wordt op foute uitgangspunten. Als de metingen de Poisson verdeling niet volgen maar wel door een analist gebruikt worden om een procesaanpassing te onderbouwen, dan maakt deze een beslissing op basis van een verkeerd uitgangspunt. Een geschikte manier om dit te voorkomen is filteren door de Kolmogorov-Smirnov test.

Op de kwaliteit, kwantiteit van de gegevens alsmede de Poisson verdelingen wordt ingegaan in sectie 4.1. Hierna wordt worden de toepassingen besproken van de Kolmogorov-Smirnov goodness-of-fit test in sectie 4.2.

4.1. Karakteristieken van de MAASTRO gegevensbron

De gegevensbron bestaat voor de MAASTRO uit een paar duizend volgens het Eindhoven Classificatie Model geclassificeerde basisoorzaken (appendix A). Dit zijn classificaties onderverdeeld in vier groepen (technisch, menselijk, organisatorisch en overig). Aan elke basisoorzaak zijn contextvariabelen toegekend (*e.g.*, sprake van avonddienst, sprake van oude apparatuur, ...). De metingen zijn over de periode medio 2005 t/m medio 2007.

Wat deze gegevensbron lastig maakt om te beheersen is dat informatie voor normalisatie afwezig is. De classificaties hebben hun oorsprong uit de PRISMA analysebomen, en deze zijn uitgevoerd op basis

[1] zie voetnoot 3, sectie 3.3.

van een melding van een incident dat heeft plaatsgevonden in een behandeling. Dit suggereert het normaliseren van de aantallen classificaties in bijvoorbeeld aantal uitgevoerde behandelingen. Dergelijke cijfers zijn niet in de gegevensbron aanwezig en zouden uit een extern systeem gehaald moeten worden. Dit maakt het lastig om de gegevens te beheersen.

De metingen in de gegevensbron van de MAASTRO volgen per week de Poisson verdeling. In subsectie 4.1.1 wordt uitvoeriger ingegaan op de Poisson verdelingen in de MAASTRO gegevensbron.

4.1.1. De metingen volgen een Poisson verdeling

De Poisson verdeling zal gebruikt worden om de analist te ondersteunen in zijn onderbouwing van verbetermaatregelen. Hiervoor is het van belang dat de PRISMA gegevens van de MAASTRO overeenkomen met de Poisson verdeling. Als dat het geval is kan de kans berekend worden dat de meting de accuraat is in die zin dat het echt de patiëntveiligheid weergeeft, en dat in feite de verhoudingen tussen typen fouten niet heel anders liggen. Als de analist opzoek is naar de werkelijke structurele problemen, en hij daarvoor een beperkt budget of beschikbare tijd heeft, is de kans groter dat efficiëntere verbetermaatregelen genomen worden. Aantallen incidenten samengenomen per week suggereren een Poisson verdeling daarom wordt deze verdeling uitgeprobeerd.

De Poisson verdeling is een kansverdeling voor discrete metingen dat het aantal events telt per tijdsinterval of bijvoorbeeld afstand. Wij gebruiken een interval van een week en regel is wel dat de events, de basisoorzaken, niet tegelijk optreden en dat ze "geheugenloos" zijn (d.w.z. het optreden van (meer of minder) events heeft geen invloed op de toekomstige metingen). Omdat specifiek geclassificeerde basisoorzaken observaties zijn van structurele (basis) problemen gebruiken die over een bepaalde periode bij elkaar opgeteld kunnen worden, gebruiken wij de Poisson verdeling per tijdsinterval *i.e.*, per week.

De eis dat events in een Poisson verdeling niet tegelijk mogen optreden wordt niet gehaald met de basisoorzaken maar dat maakt praktisch niet uit. Een classificatie Technical/Design (TD) kan in één analyse meerdere keren voorkomen en een analyse heeft betrekking op één incident waar één tijdstip aan verbonden is. Op deze manier krijgen meerdere basisoorzaken dezelfde tijd mee, ondanks dat ze in de praktijk niet simultaan hebben plaatsgevonden. Daarom is de verwachting dat dit niet niet de kwaliteit van de verdeling in de weg zal zitten.

De eis dat events "geheugenloos" zouden moeten zijn wordt wel gehaald. Elk incident, dus ook de near misses worden geregistreerd en daar zijn geen voorwaarden aan verbonden. Deze redenen zorgen ervoor dat de Poisson verdeling toegepast kan worden.

In de formule van de verdeling is de kans op precies "k" events:

$$P(N = k) = \frac{e^{-\lambda}\lambda^k}{k!}.$$

Hierin is λ een positief reëel getal, gelijk aan het aantal verwachtte voorvallen in het tijdsinterval en e is een constante, het grondtal van de natuurlijke logaritme[2] (een benadering is $e = 2.718281828459...$).

Toepassing De gegeven Poisson formule wordt toegepast op de PRISMA gegevens van de MAASTRO. Metingen worden per week gegroepeerd, een voorbeeld week is te vinden in tabel 4.1. In deze week is het aantal metingen 12, zo worden van alle weken de aantallen bij elkaar opgeteld en gedeeld door het aantal weken. Dit gemiddelde is de lambda in de Poisson formule.

Nu kunnen de daadwerkelijke metingen en de verwachtte metingen volgens de Poisson in een histogram getekend worden. Een paar honderd figuren zijn gegenereerd zoals in figuren 4.1 4.2 en 4.3. Voor alle

[2] zie de volgende website `http://mathworld.wolfram.com/e.html`.

classificaties, groepen van classificaties en individuele classificaties; voor specifieke intervallen per dag, per week en per maand. Aan de hand van de figuren is duidelijk geworden dat het juiste interval per week is.

Dit volgt uit het de figuren 4.1 4.2 en 4.3. De observaties per dag neigen zeer sterk naar nul, in zoverre dat de Poisson verdeling die bij het gemiddelde (de lambda) hoort niet meer overeenkomt. Per week is dit duidelijk niet het geval en de observaties per maand volgen de verdeling erg slecht. De rede dat per maand (en per kwartaal, jaar, ...) de verdeling niet overeenkomt is omdat ten eerste er te weinig metingen zijn wanneer over zulke lange perioden aantallen geaggregeerd worden. Ten tweede hadden we het juiste interval al gevonden per week, en Poisson verdeling hiervan gaat verloren bij grotere intervallen. De lambda gaat wel omhoog, maar de scherpte van de distributie niet, die zit er wel in bij intervallen per week.

Bestudeerde alternatieven Naast groeperen per interval zijn ook verschillende verdelingen uitgeprobeerd zoals het gebruik maken van de afstand tussen de (bijna-)incidenten (de afstanden lijken een exponentiële verdeling te volgen). Het gebruik van de afstand is lastig te combineren met de poisson verdeling. De grootheid waarin de afstand wordt uitgedrukt is essentieel (per dag, weken, maanden) of andersom bezien als de grootheid in bijvoorbeeld seconden uitgedrukt wordt zou de lambda gigantisch hoog worden in de formule, dit heeft als gevolg dat de kurtosis (kromming van de theoretische verdeling) zeer scherp wordt en dat juist terwijl ons verschil in seconden enorm groot kan zijn. Omdat de besproken toepassing beter werkte is hier niet verder op ingegaan.

Conclusie De rede dat de metingen per week wel Poisson verdeeld zijn is te verklaren omdat op bepaalde dagen(*i.e.*, weekenden) minder tot niet gewerkt wordt. Bepaalde behandelingen vinden alleen op maandag, dinsdag of woensdag plaats, bepaalde artsen hebben vrij op woensdag, *etc.* Per week worden dergelijke verschillen gecompenseerd omdat niet regelmatig voorkomt dat een week lang geen activiteit is. Per maand en per kwartaal wordt de data erg snel te dun, in plaats van 52 in het geval van weken nog maar 12 of 4 per jaar, een fatsoenlijke verdeling opbouwen wordt dan moeilijk. Dat is één van de redenen dat we de verdelingen per week opbouwen. De Kolmogorov-Smirnov zal hetzelfde aantonen in sectie 4.2.

Dat de overige intervallen niet overeenkomen is niet niet erg want de verdeling per week kan gebruikt worden. Dat is geldig vanwege de additiviteit ("additive property") van de Poisson verdeling, dat inhoudt dat een kwartaal als de som van dertien weken beschouwd kan worden.

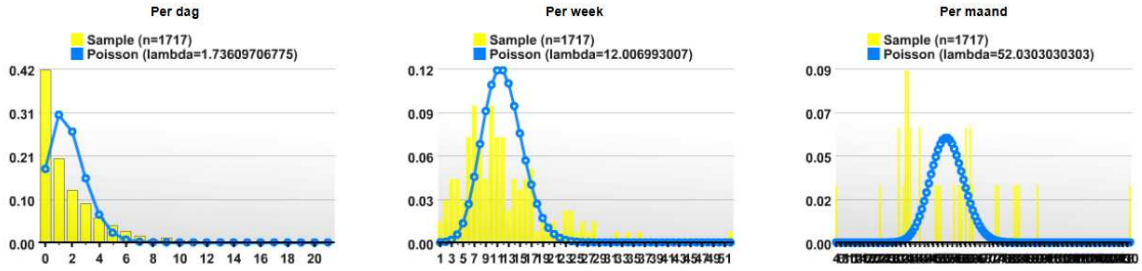
De rede dat bepaalde selecties niet Poisson verdeeld zijn is te verklaren door de stelling van A.Weber [37]. Bepaalde selecties zoals groep menselijk falsen of de classificatie HRI. De stelling van Weber luidt: "De som van twee onafhankelijke Poisson verdeelde stochasten is weer Poisson verdeeld, met als parameter de som van de parameters.". Dit wordt aangetoond in de conclusie in sectie 4.2.1, nadat de goodness-of-fit tests zijn besproken die we hierin gebruiken (sectie 4.2).

4.2. Goodness-of-fit test voor de Poisson verdeling

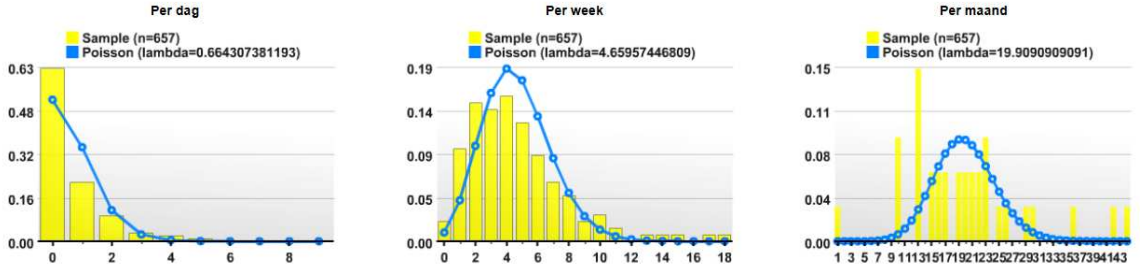
Twee goodness-of-fit tests worden uitgeprobeerd op alle classificaties en groepen daarvan en getoetst op beoordelingsvermogen. De gebruikte tests zijn de Kolmogorov-Smirnov (K-S) en de Chi-kwadraat test. Met classificaties wordt bedoelt *e.g.*, Technical/Design (TD), T-EX, *etc.*, en de toetsing wordt gedaan door de uitkomsten van de tests te vergelijken met de visualisaties (de histogrammen van de metingen met een extra lijn daar doorheen, met de verwachtte waarde volgens Poisson). Hieruit volgde dat de Kolmogorov-Smirnov (K-S) test goed werkt, en de Chi-kwadraat niet.

nummer	tijdstip	afstand tot volgend incident.
01.	maandag 12:00:00	1.66667
02.	woensdag 19:20:00	3.34722
03.	woensdag 21:40:00	0.09722
04.	donderdag 01:00:00	0.13889
05.	donderdag 10:00:00	0.37500
06.	vrijdag 16:00:00	1.25000
07.	zaterdag 01:00:00	0.37500
08.	zaterdag 11:15:00	0.42708
09.	zaterdag 11:20:00	0.00347
10.	zaterdag 14:30:00	0.13194
11.	zondag 01:00:00	0.43750
12.	zondag 01:00:00	0.00000

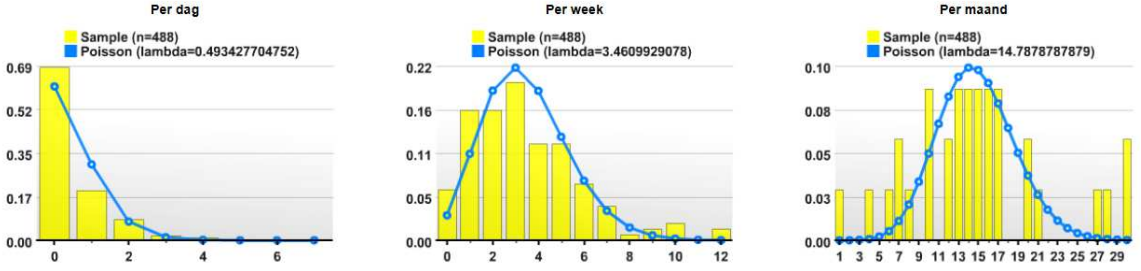
Tabel 4.1: Voorbeeld data twaalf incidenten in één week, aantal=12



Figuur 4.1: Meldingen van alle classificaties per dag, week en maand (vlnr).



Figuur 4.2: Meldingen met een technische classificatie per dag, week en maand (vlnr).



Figuur 4.3: Meldingen met TD classificatie per dag, week en maand (vlnr).

De verwachting is dat de processen overeenkomen met de Poisson verdeling. De processen binnen de radiotherapie zijn namelijk dermate gestandaardiseerd en naar verwachting stabiel, dat de verwachting is dat deze niet extreem zullen afwijken. Uit de K-S test blijkt dat dit op bepaalde plaatsen wel gebeurd.

De K-S test kan de onderliggende oorzaak van de niet-matchende classificaties niet aantonen. Zonder volledige zekerheid te kunnen bieden is de oorzaak vaak te weinig metingen, maar zou het ook kunnen dat een dermate sterke verstoring heeft opgetreden dat de test faalt, maar de gegevens in principe Poisson verdeeld zijn. Afhankelijk waar men naar op zoek is, dergelijke sterke of subtielere verstoringen, kan de K-S test gebruikt worden voor het wegfilteren van deze classificaties.

De K-S test zal nu verder besproken worden aan de hand van de gegeven Poisson verdelingen uit figuren 4.1, 4.2, 4.3. De uitkomsten van de test op alle classificaties en groepen is te vinden in appendix D.

4.2.1. De Kolmogorov-Smirnov goodness-of-fit test voor de Poisson

Achtergrond informatie De K-S formule is o.a. door A. N. Pettitt aangepast t.b.v. hantering van discrete data [20] en deze aanpassing schijnt volgens hen in bepaalde omstandigheden zelfs beter te zijn dan de (voor discrete data beschikbare) Pearson Chi kwadraat test. In twee softwarepakketten: Mathwave EasyFit 4.0 en SPSS 14.0. In EasyFit is de K-S test expliciet beschikbaar voor discrete data. De K-S test is in beginsel niet geldig voor discrete data (o.a. volgens het Engineering Statistics Handbook [38]). De formule in de EasyFit handleiding laat geen aanpassingen zien t.b.v. discrete data [3]. SPSS houdt waarschijnlijk *wel* rekening in zijn formule ten opzichte van discrete data omdat in de handleiding uitvoerig een test case wordt gedemonstreerd waar discrete data gebruikt wordt voor exact dezelfde test die nodig was. Daarbij komt dat de uitkomst ook anders was dan in de uitkomst uit Easyfit, en deze gebruikte een onaangepaste formule.

Toepassing Drie selecties zijn handmatig uit de database in de programma's gezet en handmatig zijn de toetsen uitgevoerd met de twee tools. Op alle selecties is inmiddels een goodness of fit test uitgevoerd in SPSS, zie tabellen in appendix D. We gaan voor nu ook weer alleen in op de voorbeeld histogrammen, de K-S test resultaten voor de voorbeeld histogrammen zijn te vinden in: 4.2, 4.3, 4.4. Toelichting op deze tabellen: de "Easyfit statistiek."-kolom is Kolmogorov-Smirnov statistiek D (zie vorige voetnoot) berekend door Easyfit. de "SPSS statistiek." kolom is het (Kolmogorov-Smirnov)Z-test statistiek berekend door SPSS [4] In Easyfit "approximates" (i.v.m. formule voor continue verdelingen) de p-waarde *misschien*. De p-waarden zijn in beide programma's *juist* significant indien >0.05, dit in tegenstelling tot alle overige plaatsen in deze scriptie. Aangegeven is in de kolommen "Easyfit resultaat." en "SPSS resultaat" of de softwareprogramma's de nulhypothese accepteren of niet.

[3] De volgende formule staat in de Mathwave Easyfit 4.0 documentatie.
"Kolmogorov-Smirnov Test:
This test is used to decide if a sample comes from a hypothesized continuous distribution. It is based on the empirical cumulative distribution function (ECDF). Assume that we have a random sample x1, ... , xn from some distribution with CDF F(x). The empirical CDF is denoted by

$F_n(x) = \frac{1}{n} \cdot [Number\ of\ observations \leq x]$

Definition:
The Kolmogorov-Smirnov statistic (D) is based on the largest vertical difference between the theoretical and the empirical cumulative distribution function:

$D = \max_{1 \leq n} \left(F(x_i) - \frac{i-1}{n}, \frac{i}{n} - F(x_i) \right)$"

[4] Uit de documentatie van SPSS: "The Z test statistic is the product of the square root of the sample size and the largest absolute difference between the empirical and theoretical CDFs.".

Conclusie De onaangepaste K-S formule (m.b.t. discrete metingen)—de "Easyfit resultaten" kolom—komt vaak in de buurt maar is bij deze aangetoond als onbetrouwbaar (als *approximation*). In tabel 4.4 is de nulhypothese voor het interval per week niet geaccepteerd, terwijl deze wel lijkt te passen op basis van handmatige.

De K-S test in SPSS —kolom "SPSS resultaten"—geeft echter perfecte resultaten. Als "menselijke beoordelaar" ben ik het eens met alle oordelen, met uitzondering van de goedkeuringen van de maand en kwartaal histogrammen. Deze zijn echter niet relevant omdat een zo klein mogelijk correct interval (per week [5]) wenselijker is (beter veel dan weinig metingen). Mogelijk geeft dit aan dat ondanks dat de data te dun is, het niet correct is om een groter interval te selecteren dan mogelijk. Alle verdelingen die *niet* poisson verdeeld zijn detecteert de test per week goed.

Uit de Kolmogorov-Smirnov tests blijkt verder dat een interval per dag slecht overeenkomt met de Poisson verdeling ondanks dat voor het menselijk falen dit lijkt mee te vallen (zie figuur 4.1 vs. 4.2). In subsectie 4.1.1 is duidelijk geworden wat de onderliggende redenen waren voor deze slechte match.

Conclusie De categorie menselijk falen is volgens de K-S test niet Poisson verdeeld omdat twee "sub selecties" (of processen), de classificaties HRI en HRV niet Poisson verdeeld waren. Het één voor één weghalen van de classificaties uit de verzameling menselijk falen, zie tabel 4.4, zorgt ervoor dat de K-S test uiteindelijk slaagt.

	N	Poisson Parameter(a,b)	Most Extreme Differences			Kolmogorov-Smirnov Z	Asymp. Sig. (2-tailed)
		Mean	Absolute	Positive	Negative		
menselijk	143	8,2657	0,173	0,173	-0,104	2,073	**0,000**
-HRI	143	5,9860	0,155	0,155	-0,092	1,856	**0,002**
-HRI -HRV	141	3,3191	0,113	0,113	-0,097	1,344	0,054

Figuur 4.4: K-S test op Menselijk falen na weglaten HRV en HRI.

Deze observatie is belangrijk omdat het betekend dat wanneer een classificatie niet Poisson verdeeld is, verder in het drilldown proces de verdeling wel aanwezig is. Datgene dat wegvalt tijdens de drilldown kan namelijk de verstoring hebben veroorzaakt.

interv.	N	lambda	Easyfit stat.	EasyFit resultaat	SPSS stat.	SPSS resultaat
dag	989	1.7361	0.23633	REJECTED 0.00000 <0.05	7,43223	REJECTED 0.00000 <0.05
week	141	12.007	0.22549	REJECTED 0.0000007 <0.05	2,69642	REJECTED 0.00000 <0.05
maand	33	52.03	0.40401	REJECTED 0.0000022 <0.05	2,32084	REJECTED 0.00000 <0.05

Tabel 4.2: Uitkomsten K-S test alle classificaties. N = aantal intervallen, *i.e.*, weken, *etc.*

interv.	N	lambda	Easyfit stat.	EasyFit resultaat	SPSS stat.	SPSS resultaat
dag	989	0.66431	0.51463	REJECTED 0.00000 <0.05	3,56238	REJECTED 0.00000 <0.05
week	141	4.6596	0.11504	ACCEPTED 0.0441 ≈ 0.05	1,34293	ACCEPTED 0.05427 >0.05
maand	33	19.909	0.2041	ACCEPTED 0.11073 >0.05	1.17244	ACCEPTED 0.12792 >0.05

Tabel 4.3: Uitkomsten K-S test technische falen (T-EX, TD, TC, TM). N = aantal intervallen, *i.e.*, weken, *etc.*

[5] Per week is elementair gebleken in sectie 4.1.1.

interv.	N	lambda	Easyfit stat.	EasyFit resultaat	SPSS stat.	SPSS resultaat
dag	989	0.49343	0.61053	REJECTED 0.00000 <0.05	2.35897	REJECTED 0.000003 <0.05
week	141	3.461	0.155	REJECTED 0.00201 <0.05	1.03166	ACCEPTED 0.23759 >0.05
maand	33	14.788	0.14358	ACCEPTED 0.46187 >0.05	0.82482	ACCEPTED 0.50435 >0.05

Tabel 4.4: Uitkomsten K-S test classificatie technisch design (TD). N = aantal intervallen, *i.e.*, weken, *etc.*

5 Data-mining PRISMA gegevensbron

Dit hoofdstuk beantwoord subonderzoeksvraag Sub-2 (opgedeeld):

- Welke informatie kan met behulp van data-mining technieken uit dezelfde gegevensbron gehaald worden?
- Wat is daarvan geschikt voor benchmarking?
- Hoe kunnen we valideren of de uitkomsten voor het plegen van interventies (procesaanpassingen) bruikbaar zijn?

Data-mining is "the nontrivial extraction of implicit, previously unknown, and potentially useful information from data" [1]. In ons geval is het naast vergelijken van PRISMA profielen—sectie 2.3.2 en figuur B.4—interessant om te proberen deze "impliciete, voorheen onbekende en potentieel bruikbare informatie" te extraheren.

Een op N-grams gebaseerd algoritme kan uit de gegevensbron nog meer informatie beschikbaar stellen voor analyse en benchmarking. Namelijk combinaties zoals de PRISMA classificatie Technical/Construction waarvan sprake was van avonddienst, oude apparatuur en jong personeel (contextvariabelen). Het aantal keer dat deze combinatie voorkomt binnen alle instituten is een stuk informatie dat vergeleken kan worden met alle andere combinaties, om zo de "dominerende" combinaties op te sporen. Daarnaast kan deze combinatie per instituut uitgerekend worden en in benchmarking gebruikt worden om deze te vergelijken. Op deze manier wordt er door data-mining nog meer informatie beschikbaar voor analyse en benchmarking.

K-Means clustering kan op dit moment niet van worden aangetoond dat het bruikbare informatie voor de analist oplevert. Voor het clusteren van PRISMA analysen worden de classificaties en contextvariabelen gebruikt. Hieruit volgt dat het algoritme werkt, echter door de resultaten te interpreteren, deze te verklaren en ze later nog te visualiseren moet geconcludeerd worden dat voor nu niet aangetoond kan worden dat K-Means clustering ook echt bruikbare informatie oplevert voor de analist.

De tekstuele omschrijvingen van de basisoorzaken vormen een interessante bron van informatie ondanks de kwaliteit van de tekst en ondanks het feit dat de radiotherapie hier niet van plan is iets mee te doen. Alle tekst over één jaar is samen voor één instituut al 267KB (wat gelijk staat aan 273408 karakters en dus heel wat woorden). Dit zijn veel meer woorden dan bijvoorbeeld aantallen classificaties of contextvariabelen. De kwaliteit van de tekst is slecht in die zin dat niet altijd netjes woorden voluit worden geschreven.

In het Information Retrieval (IR) vakgebied zijn methoden ontwikkeld om te compenseren voor slechte kwaliteit tekst. Een van de toepassingen van IR op documenten die ingescand zijn middels OCR, bevatten vaak letters die niet kloppen (een i wordt een l, of een 1, *etc.*). Als de zoekterm "lampion" is en een woord verkeerd ingescand is als "Iampion" kan N-Grams ervoor zorgen dat het woord op één letter na wel als match herkent wordt. Dit is een voorbeeld hoe gecompenseerd kan worden voor slechte kwaliteit tekst.

[1] W. Frawley and G. Piatetsky-Shapiro and C. Matheus (Fall 1992). "Knowledge Discovery in Databases: An Overview". AI Magazine: pp. 213.228. ISSN 0738-4602.

Het probleem met subonderzoeksvraag Sub-2 is dat we weten hoe validatie mogelijk is, maar dit tot op heden nog niet mogelijk is geweest. Dit geldt ook voor onder andere de uitkomst van het opensource programma SemanticEngine, dat gebruikt wordt om Latent Semantic Indexing uit te proberen op de tekst uit de MAASTRO gegevensbron.

Naast het genoemde K-Means en N-grams algoritme proberen we het opensource programma CompLearn ook uit op de gegevensbron, in sectie 5.1. Het opensource programma SemanticEngine wordt uitgeprobeerd op de gegevensbron, en onderzocht wordt of een Information Retrieval systeem uit de pathologie te gebruiken is op onze gegevensbron, in sectie 5.2.

5.1. Clustering analyse

Om echt tot de structurele problemen te komen is het belangrijk dat we niet ons blijven richten op één kant van het verhaal. In de PRISMA systematiek wordt gekeken naar bepaalde typen oorzaken (*i.e.*, T-EX (technisch extern)) of specifiekere typen oorzaken (*i.e.*, T-EX in speciale omstandigheid avonddienst). Gezegd kan worden dat de typen oorzaken bestaan uit een aantal factoren: de contextvariabelen en classificaties. De bedoeling is daar de structurele problemen uit te halen en met dat inzicht aanpassingen te maken in het proces of een andere actie te ondernemen. Hier wordt nog niet goed rekening gehouden met het feit dat *combinaties* van factoren belangrijker kunnen zijn dan de factoren los van elkaar.

Voor een analist is het over het hoofd te zien dat een bepaalde factor A (avonddienst) in combinatie met een andere factor B (nieuw personeel) veel vaker voorkomt dan de schijnbaar dominante factor C (nieuw apparaat). Voor dit probleem is een simpel combinatie-zoek-functie al voldoende.

Naast het vinden van dat soort combinaties kan clusteringanalyse, indien goed toegepast, nog meer verbanden aantonen die de analist kunnen ontgaan. Clustering-analyse kan namelijk gegeven de combinaties ook kijken welke daarvan weer het meest op elkaar lijken, en door deze te groeperen, kan onderzocht worden welke groepen in *dat* geval dominant aanwezig zijn in de gegevensbron. De rede dat de analist deze verbanden niet kan herkennen is de complexiteit en de hoeveelheid metingen die onderzocht moeten worden. De Clusteringanalyse zal worden uitgevoerd aan de hand van een K-Means implementatie, daarna wordt kort een toepassing van clustering op compressie (benadering van de Kolmogorov-Complexiteit) met behulp van een open-source tool CompLearn uitgeprobeerd en besproken. De motivatie voor het K-Means algoritme is geweest de gemakkelijke implementatie. De motivatie voor CompLearn is dat een toepassing door een "off the shelf" tool snel te verifiëren is [15].

5.1.1. Clustering met K-Means

K-Means is een clustering algoritme dat gebruikt kan worden om objecten te groeperen op basis van attributen of *kenmerken* die de objecten omschrijven (karakteristieken). Een object kan gezien worden als een vector in een multidimensionale ruimte waarin elk karakteristiek een dimensie vertegenwoordigd. De K-means formule kan uitgedrukt worden in een functie voor de kwadratische fout (euclidische afstand), en richt zich op het minimaliseren van de variatie in de clusters (referentie, zie voetnoot[2]):

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2$$

De formule drukt uit dat voor elke cluster S_i het volgende is uitgerekend: de som van de kwadratische verschillen van elke vector *in* die cluster, x_j, met de centroïde van die cluster, μ_i. De kwadratische fout

[2] De K-Means formule is overgenomen uit `http://en.wikipedia.org/wiki/K-means_algorithm`. De formule komt overeen en is geïmplementeerd aan de hand van `http://people.revoledu.com/kardi/tutorial/kMean/Algorithm.htm` (website van Dr. K. Teknomo), hieruit zijn de stappen van Lloyds algoritme ook overgenomen.

is een formule voor de euclidische afstand zonder de worteltrekking, deze is in feite ook irrelevant voor de afstandsberekening.

Belangrijke aannamen in het gebruik van het algoritme zijn dat de data natuurlijk geclusterd is en dat de clusters "sferisch" ("cirkelvormig") zijn. Enkele nadelen zijn: k moet op voorhand bepaald worden en de initiële plaatsing van centroïden heeft invloed op de uitkomst. Het algoritme kan meerdere keren uitgevoerd worden om dit probleem te minimaliseren [3].

Om precies te zijn is Lloyd's algoritme toegepast om het K-Means "probleem" op te lossen. Het algoritme in vier stappen omschreven (referentie, zie voetnoot [4]):

Stap 1 Initialiseer k centroïden in de vectorruimte. Dit kunnen de eerste k metingen zijn, random posities of de uitkomst van een eerdere "run" van het algoritme.

Stap 2 Plaats elke vector, X_j, in de groep waarvan de centroïde, μ_i, het meest dichtstbijzijnd is.

Stap 3 Als alle vectoren zijn toegewezen, herbereken de posities van de K centroïden.

Stap 4 Herhaal stap 2 en 3 totdat er geen beweging meer zit in de centroïden.

Toepassing In een voorbeeld wordt de formule concreet toegepast op een klein proces proces genaamd "fysisch lab". De periode is 2006. In de K-Means formule is k op 3 ingesteld en dit resulteert in drie centroïden: μ_1, μ_2 en μ_3. Omdat het aantal aanwezige PRISMA analysen dat betrekking heeft gehad op dit proces 37 is, zijn er de vectoren: x_1, x_2, ..., x_{37}.

De vier stappen uit het algoritme kunnen nu uitgevoerd worden. Na initialisatie (stap 1) en net zo lang passen en meten (stap 2 en 3), is het clusteren klaar indien er geen beweging meer zit in de groepen (stap 4). In tabel 5.1 is te zien hoe de vectoren (3 van de 37) en alle centroiden eruit zien over alle dimensies (contextvariabelen), na het uitvoeren van de stappen. In elke iteratie van het algoritme zijn de vectoren toegewezen aan de dichtstbijzijnde cluster, de centroid wordt als gemiddelde in het midden geplaatst en dit zijn de waarden die staan in de tabel. Na deze (her)berekening van de centroïden worden opnieuw de vectoren toegekend aan de dichtstbijzijnde centroid, *etc.* Op het moment dat hier geen beweging meer in zit stopt het algoritme met verfijnen van deze waarden.

Een uiteindelijke verdeling van vectoren over de clusters is voor de drie voorbeeld vectoren te zien in tabel 5.2. Zo is in de rij voor vector x_1 bijvoorbeeld te zien dat de afstand tot cluster S_1 in dit geval het kleinst is, en daar wordt die vector dus aan gekoppeld.

Visualiseren van de clusters voor de analist De clusters worden op twee manieren gepresenteerd, waarvan de eerste is voor elke cluster een lijst van combinaties contextvariabelen. Dat houdt in, gesorteerd op frequentie van voorkomst, de combinaties van contextvariabelen. Daarmee kunnen we namelijk antwoord geven op *e.g.*, de volgende vraag: "In de PRISMA analysen van 2006 met een T-EX classificatie voor de cluster `G1`; welke combinaties van contextvariabelen zitten daarin inclusief de combinaties die *daar* weer de sterkste overeenkomst mee hebben, resulteren in die specifieke cluster?" Deze presentatie geeft de analist hopelijk een beter idee van het typen oorzaken dat de cluster karakteriseert.

[3] "Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centres. The k-means algorithm can be run multiple times to reduce this effect." bron: `http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html`.

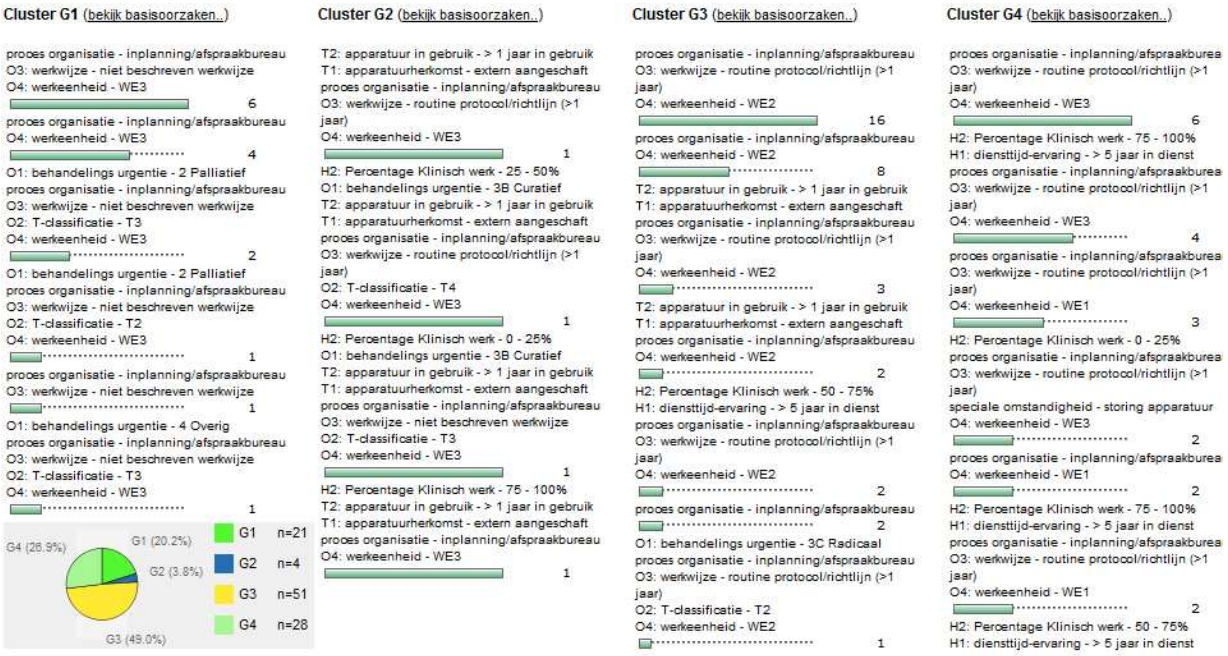
[4] *ibid.*, 2.

Contextvariabelen (dimensies)	PRISMA analysen (vectoren)			Centroids		
	x_1	x_2	x_3	μ_1	μ_2	μ_3
H2: Percentage Klinisch werk - 0 - 25%	1	0	0	0.18987	0	0.01875
H1: diensttijdervaring - 1-5 jaar in dienst	0	1	1	0.04430	0.80263	0.0375
O1: behandelings urgentie - 2 Palliatief	0	0	0	0.05696	0.02631	0.0875
H2: Percentage Klinisch werk - 25 - 50%	0	0	0	0.09494	0	0.01875
O1: behandelings urgentie - 3A Curatief	0	1	1	0.05063	0.76316	0.39375
O1: behandelings urgentie - 3B Curatief	0	0	0	0.18987	0.01316	0.09375
O1: behandelings urgentie - 3C Radicaal	0	0	0	0.00316	0.02632	0.08125
O1: behandelings urgentie - 4 Overig	0	0	0	0.00316	0	0
O1: behandelings urgentie - 5 Planbaar bestralen	0	0	0	0	0	0
H2: Percentage Klinisch werk - 50 - 75%	0	0	0	0.08228	0.03947	0.01875
O1: behandelings urgentie - 6 Uitzonderingen	0	0	0	0.02848	0	0
H2: Percentage Klinisch werk - 75 - 100%	0	1	1	0.07278	0.98684	0.5
O3: werkwijze - ¡ 1 jaar bestaand protocol/richtlijn	0	0	0	0.08544	0	0.09375
H1: diensttijdervaring - ¡ 1 jaar in dienst	0	0	0	0.03165	0	0.0125
T2: apparatuur in gebruik - ¡ 1 jaar in gebruik	0	0	0	0.16456	0.02632	0.575
...	...	...	...	...	...	...

Tabel 5.1: Berekende afstanden in alle dimensies voor elke centroid, met 3 van de 37 vectoren als voorbeeld.

Clusters (k=3)	PRISMA analse vectoren		
	x_1	x_2	x_3
S_1	**2.191062091**	2.589559108	2.540207042
S_2	3.103980556	**1.148436249**	**1.113534215**
S_3	3.039621173	2.650810607	2.537576969

Tabel 5.2: De wijze waarop de drie vectoren uit tabel 5.1 in clusters geplaatst zijn.



Figuur 5.1: MAASTRO proces inplanning/afspraakbureau 2006. k=4, n=177, clusters op contextvariabelen en keuzen. Weergave in clusters is *aantal voorgekomen (exacte) combinaties*.

In figuur 5.1 is dit in een voorbeeld te zien, in cluster G3 was 16 keer sprake van de volgende combinatie: "een analyse waarbij sprake was van het proces inplanning/afsprakenbureau, een routine/protocol dat ouder is dan een jaar en waarbij de werkeenheid WE2 is." De bedoeling is dat als er naast deze combinatie, allerlei combinaties voorkomen die erop lijken, de analist verbanden kan zien en kan concluderen dat *e.g.*, de cluster voornamelijk oorzaken omvat die optreden tussen een specifieke werkeenheid en proces.

Deze visualisatie werkte niet omdat de contextvariabelen nog niet toereikend zijn. Ze worden niet goed ingevuld en het is sowieso lastig te interpreteren, waarschijnlijk vanwege de complexiteit. De complexiteit zit hem in het feit dat het vaak toch een hele lijst aan combinaties is, en dus niet makkelijk te herkennen is wat die lijst precies voorsteld. Daarom is deze visualisatie niet zinvol gebleken.

Het combinatie algoritme is echter wel op een andere manier zinvoller te gebruiken. Namelijk door te clusteren op één cluster, oftewel het niet conceptueel opdelen van de totale groep. Dit zorgt ervoor dat deze manier van visualiseren antwoord geeft op de volgende vraag: "Welke combinaties van contextvariabelen komen samen het meest voor (in PRISMA analyses), en veroorzaken in een specifieke periode een bepaalde classificatie T-EX?" Hiermee is in feite een oplossing gevonden voor de gewenste nieuwe "rapportages", die precies antwoord geven op deze vraag.

Visualiseren voor beoordelen clusters De vorm van de clusters wordt gevisualiseerd in een scatterplot in 2d en 3d, in SPSS 14.0, om de *cluster cohesion* (compactness, tightness) en cluster seperation (isolation) [29, p. 535] inzichtelijk te krijgen. Elke PRISMA analyse krijgt een cirkel met een specifieke kleur die hoort bij de cluster waar deze in is geplaatst volgens K-Means.

De verschillen (of overeenkomstigheden) tussen de analysen op basis van de contextvariabelen bepalen de afstanden tussen de analysen. In het geval van een 3d scatterplot worden *e.g.*, de technische contextvariabelen samengenomen en deze worden gebruikt voor de spreiding over de X-as, de organisatorische voor de Y-as, menselijke en overige over de Z-as.

Hoe de clusters ten opzichte van elkaar bij elkaar horen–of door elkaar heen lopen— wordt gevisualiseerd met de open source tool CompLearn. Een distance-matrix wordt gegenereerd in een formaat dat gelezen wordt door CompLearn's maketree programma. In de matrix staat voor elke analyse de berekende (euclidische) afstand tot elke andere analyse. Deze genereert op een brute-force manier de "best-fitting" unrooted binary tree, dat wil zeggen een "boom" structuur zonder root-node waarvan elke node maximaal twee childs mag hebben. Hier komt een bestand uit dat graphviz kan visualiseren, in dat bestand is met een script kleur achteraf toegevoegd aan elke node op basis van welke clusters deze in zit.

Conclusie De waarde van de K-Means implementatie is niet aangetoond. De hoop was dat deze wel of niet aangetoond kon worden in de evaluatie van het prototype, zie sectie 3.4.1. De rede dat we niet kunnen aantonen dat het werkt is het gebruik van en de kwaliteit van de contextvariabelen. Omdat de contextvariabelen reeds bijgesteld zijn, omdat de analisten al doorhadden dat de contextvariabelen niet ideaal gekozen waren, kan nog geen definitief uitsluitsel gegeven worden dat K-Means op contextvariabelen wel of niet zin heeft, en dus toegevoegde waarde heeft. Dit kan in de toekomst nog blijken, echter voor nu moet geconcludeerd worden dat K-Means geen zin heeft.

5.1.2. Zoeken naar contextvariabelen combinaties met N-Grams

Het kan zijn dat een combinatie van contextvariabelen zeer vaak voorkomt maar overschaduwd wordt door andere contextvariabelen. In die zin dat het steeds dezelfde combinatie is maar die de ene keer optreedt in het ene proces en de andere keren in een ander proces. Het idee van *n-grams*[5], dat in

[5] url `http://en.wikipedia.org/wiki/N-gram`

text-analyse onder andere gebruikt wordt voor "approximate matching", kan gebruikt worden om deze overschaduwing tegen te gaan, en zorgt ervoor dat de analist minder over het hoofd kan zien.

Een op N-Grams gebaseerd algoritme is bedacht (appendix F, figuur F.3 [6]), dat de genoemde "approximate matching" mogelijk maakt op de contextvariabelen.

Concreet zoekt het algoritme *e.g.*, in alle combinaties *i.e.*, "A, A, B", "C, A, A", ..., ook naar "A, A", "A, B", ... Het *pair* "A, A" komt twee keer voor in de volledige combinaties en zou anders niet meegeteld worden.

In combinatie met K-Means is het vooral een verrijking voor het zoeken van combinaties binnen één cluster. Dit om de eerdere constatering dat de verschillende clusters niet te interpreteren zijn, en het ophakken van deze combinaties dan weinig toegevoegde waarde zou hebben en zelfs een vertekend beeld kan geven. Stel dat een combinatie "A, B, C" enorm veel voorkomt, maar in de clustering verdeeld is over meerdere clusters, zal deze niet meer opvallen. In dat geval is het beter om niet te clusteren maar alleen te zoeken op combinaties.

Conclusie N-grams en K-Means clustering maken de kans groter dat de analist beter geïnformeerd is voor het doen van efficiënte verbeteringen. Voor N-grams ligt het meer voor de hand dat dit nuttige informatie oplevert dan voor K-Means, omdat combinaties van contextvariabelen wel degelijk wat zeggen, waar dit voor clusters, combinaties die op elkaar lijken, helemaal niet zeker is. De efficiëntie wordt bereikt omdat de analist beperkte tijd en resources heeft en daarom moeten keuzen gemaakt worden, indien de informatievoorziening vollediger kan zijn, zoals aangetoond, wordt de kans groter dat de meest efficiënte keuze gemaakt kon worden.

5.1.3. Clustering op "compressie"

Van een stuk informatie zoals muziek, grafische bestanden of documenten is de kortste binaire beschrijving in enen en nullen de kolmogorovcomplexiteit. Een conceptueel voorbeeld, de reeks 1010101010101 010 kan opgeschreven worden als "acht keer 10", voor een volgende reeks wordt het lastiger om dit kort te formuleren: 0010011000110111 (een omschrijving zou misschien net zo lang worden als de reeks zelf). "De Kolmogorovcomplexiteit is in de praktijk onberekenbaar. Maar het bijzondere is dat er toepassingen mogelijk zijn waarvoor goede compressieprogramma's een resultaat leveren dat waarschijnlijk nauwelijks afwijkt van de theoretische Kolmogorovwaarde.". [15] [7]

De opensource tool CompLearn[8] probeert eerst de Kolmogorov Complexiteit te benaderen d.m.v. compressie, berekend daarna de afstanden tussen de bestanden en voert clustering uit. Voor elk tekstbestand A wordt een gecomprimeerde versie gemaakt, daarna wordt gekeken hoe A zich verhoudt tot elk overig tekstbestand (B, C, D, ...) wat betreft afstand. De afstand wordt bepaald door ook A+B, A+C, A+D, *etc.*, samen te comprimeren. Hoe minder de gecomprimeerde combinatie in beslag neemt *i.e.*, A+B wordt een stuk kleiner dan A+C, en dus liggen die wat betreft complexiteit dichter bij elkaar. Op basis

[6] In figuur F.2 is te zien voor k=1 hoe de paren van vijf scoren, in het prototype kunnen vervolgens paren van vier, drie t/m twee bekeken worden.

[7] De manier waarop de Kolomogorov complexiteit o.a. wordt toegepast is m.b.v. Vitányi en Li's "universele maat van gelijkenis". Als muziek de input is, van Chopin, Bach, Mendelssohn en Debussy, worden alle bestanden eerst gecomprimeerd. Daarna wordt alles paarsgewijs met elkaar vergeleken in hoeverre de gecomprimeerde bestanden op elkaar lijken. "Hoe meer twee bestanden op elkaar lijken, hoe minder ruimte nodig is om de combinatie van beide op te slaan. Tenslotte drukt het programma in een getal tussen 0 en 1 uit hoezeer het ene op het andere bestand lijkt.". Het blijkt dat het programma met deze getallen heel goed muziek van Bach en Chopin kan opdelen in clusters, en ook laat zien dat Debussy meer met Chopin verwant is dan met Bach. Ten tijde van de SARS uitbraak kon het programma laten zien dat SARS sterk verwant was met het Corona-229-virus, "een conclusie die de biologen met in jarenlang opgebouwde kennis van virussen ook stelden".

[8] Url: `http://www.complearn.org/`

van deze informatie, dat in een "distance matrix" wordt opgeslagen, gaat de tool over tot clustering.

We willen antwoord op de vraag of *i.e.*, de technische classificaties ook daadwerkelijk dicht bij elkaar liggen, voor wat betreft afstand op basis van compressie. Als dit het geval is kunnen we ervan uitgaan dat de complexiteitbenadering voor technische fouten daadwerkelijk anders is dan bijvoorbeeld organisatorische. Als dit niet het geval is kunnen we de "distance matrix" als een profiel gebruiken om te kijken welke instituten op elkaar lijken of juist niet. Daarom proberen we dit uit op de PRISMA gegevensbron.

Toepassing Voor elk type basisoorzaak, dus voor elke 21 classificaties, is een tekstbestand aangemaakt (TD.txt, T-EX.txt, *etc.*). In dit tekstbestand zijn alle basisoorzaakomschrijvingen, als platte tekst onder elkaar neergezet. Dit zijn de teksten die uit de originele PRISMA analyseboom komen. Dit resulteerde in 21 bestanden, met ongeveer in totaal 2000 omschrijvingen van één of twee zinnen, verdeeld over deze bestanden.

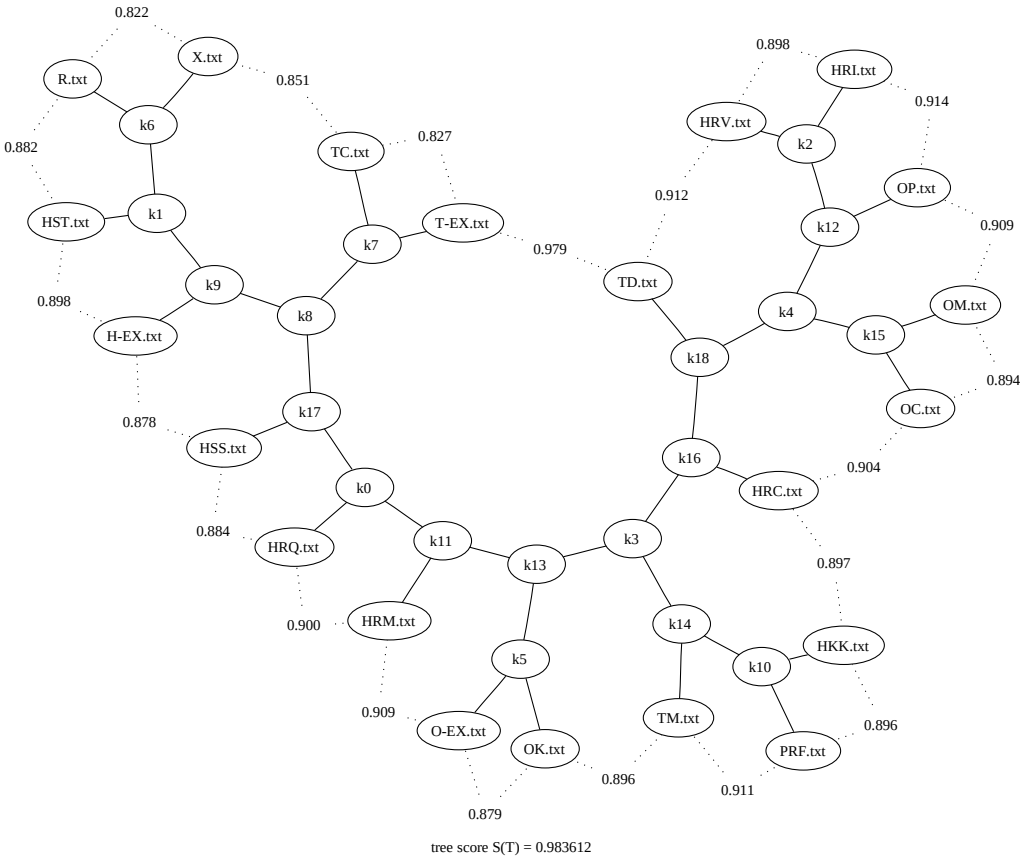
Het resultaat van de clustering van onze tekstbestanden is te vinden in figuur 5.2. Dit is een "unrooted binary tree", dat houdt in een graph zonder root-node, en waarvan elke knoop maximaal twee child nodes mag hebben. Hierin zijn de tekstbestanden de nodes (cirkels) aan de uiteinden, de beschrijving zijn de namen van de tekstbestanden. De nodes die alle tekstbestanden aan elkaar knopen, k1, k2, *etc.*, zijn nodig om de binaire boom op te bouwen (het aantal knopen is $n - 2$, in het geval van $n = 21$ classificaties daarom 19 knopen). Tussen de knopen zijn getallen zichtbaar op stippellijnen, deze geven weer hoe ideaal de afstand tussen die twee knopen in de graph wordt weergegeven. Dit getal is over een schaal van 1, waar 0 het slechts en 1 de ieale weergave inhoudt. De ideale weergave zou door de constraints van de graph in de praktijk niet gehaald kunnen worden *i.e.*, knopen kunnen niet over elkaar heenvallen, maximaal twee childs per knoop, *etc.* De treescore is ook zo'n getal op een schaal van 1, maar dan voor de hele tree, details over hoe dit te berekenen is in [3].

Conclusie Een aantal interessante observaties kunnen aan de hand van figuur 5.2 gemaakt worden. Het resultaat kan dus goed dienen als centraal onderwerp in een brainstormsessie. Ten eerste clusteren de groepen bijvoorbeeld niet netjes bij elkaar, vaak zwerven een of twee classificaties aan de andere kant van de graph. De classificatie Technical/Material (TM) bevindt zich uitzonderlijk laag ten opzichte van de overige technische classificaties. En suggereert ook het dichtsbij patiëntgerelateerde factoren (PRF) te staan.

Een tweede interessante observatie is dat de menselijke factoren allemaal aan één kant clusteren, met uitzondering van de classificaties HRV en HRI. Dit is interessant omdat dit precies de twee classificaties waren die de ervoor zorgden dat de groep menselijke factoren in zijn geheel niet Poisson verdeeld meer was, zie conclusie sectie 4.2.1.

In één oogopslag kan zichtbaar gemaakt worden hoe op basis van complexiteit de instituten zich tot elkaar verhouden. De eerder genoemde toepassing om de "distance matrix" (dat als input diende voor het genereren van de graph) te gebruiken om gelijksoortige instituten te vinden is een mogelijkheid. Maar wellicht is een directere toepassing mogelijk zoals, alle tekst simpelweg per instituut in *één* tekstbestand plaatsen en deze te clusteren. Hiervoor is wel noodzakelijk dat aangetoond kan worden dat overeenkomst op basis van compressie daadwerkelijk weerspiegeld hoe instituten qua inrichting van processen of typen fouten op elkaar lijken.

Een verdere beperking in het interpreteren van de grafieken is dat niet aangetoond kan worden waarom precies twee bestanden bij elkaar horen. In de zin van complexiteit, omdat compressie toegepast wordt is het niet mogelijk om te achterhalen welke gedeelten overeenkomen. Dit is wel mogelijk wanneer geclusterd wordt op contextvariabelen, of concepten in text-analyse, omdat dan inzichtelijk is wat precies overeenkomt. Daarom kunnen minder gedetailleerde beweringen gedaan worden over de uitkomst.



Figuur 5.2: Profiel classificaties op basis van basisoorzaakomschrijvingen.

5.2. Text-mining analyse

5.2.1. Information retrieval systeem pathologie

Het is interessant om te onderzoeken hoe informatie uit een ongestructureerde bron van vrije tekst gehaald kan worden middels tekst-analyse. Slechts enkele instituten, best-practices, hebben een volledig- of deels volledig- gestandaardiseerde dataset (paragraaf 2.2, "definitie gegevensbron"). Het gros van alle zorginstellingen registreert meldingen en de resultatieve dataset is niet-gestandaardiseerd van aard.

L.M.M. Braun, A. Hasman *et.al.* [7] hebben onderzoek gedaan naar een methode om pathologie te ondersteunen in het codificeren/classificeren van rapporten, hierin worden text-mining technieken toegepast [4]. Prof. dr. F. Wiesman is gespecialiseerd in ontologieën en heeft ook gedeeltelijk samengewerkt met L.M.M. Braun. In dit onderzoek wilden we kijken of deze text-mining techniek ook toepasbaar is op meldingen (ook een vorm van vrije tekst[9]). In beide systemen staat ook hetzelfde probleem centraal: extractie van classificatiecodes. Text-mining had in het geëvalueerde pathologie systeem als doel pathologen te ondersteunen in het classificeren van rapporten. Text-mining zou in ons systeem het doel kunnen hebben de meldingen van verschillende instanties alsnog te kunnen classificeren.

[9]Op dit moment was het onderzoek nog niet toegespitst op de Radiotherapie en het gebruik van de PRISMA methode.

Potentiële toepasbaarheid op meldingen? Het pathologie systeem was in 80% van de gevallen behulpzaam en gaf in 50% zelfs een volledig correcte suggestie (in de top-5 best gevonden classificaties). Het systeem was niet sterk genoeg om voor de pathologen volledig automatisch de rapporten te classificeren. De toepassing van een dergelijk systeem kan voor ons daarom ook alleen ondersteunend zijn. Het zou daarom als een niet waterdichte toevoeging aan de gebruiker gepresenteerd moeten worden.

Op basis hiervan en een korte verdieping in ontologieën op het internet (*i.e.*, [21, 1, 9]) ben ik met Prof. Dr. F. Wiesman in gesprek gegaan met de volgende abstracte toepassingsmogelijkheden in mijn achterhoofd. Met name het toepassen van berekende similariteit tussen meldingen ten behoeve van:

- Het vinden van soortgelijke incidenten. Op basis van een of meerdere incidenten als input. Dit zou een hulpmiddel kunnen vormen in het vinden van benchmarkpartners die bijvoorbeeld soortgelijke incidenten hadden, en inmiddels een aantal interventies hebben gepleegd.
- Het vinden van structurele problemen. Op basis van de berekende vectoren clusters extraheren (*i.e.*, m.b.v. K-Means clustering, zie subsectie hoofdstuk 5.1.1). Vervolgens kunnen wanneer de juiste labels aan deze clusters gehangen worden (*i.e.*, de drie classificatiecodes die het grootste raakvlak hebben met alle codes in de cluster) problemen aan het licht brengen die niet opvallen per zorginstelling individueel, maar wel wanneer landelijk de gegevensbronnen gecombineerd worden.

Resultaten discussie Het pathologie systeem maakte gebruik van een collectie van 7500 handmatig gecodificeerde rapporten (met in totaal ongeveer 20.000 verschillende woorden). In het pathologie systeem waren de rapporten door pathologen en specialisten geschreven, vaak ook collega's. Dit had als gevolg dat gebruikte terminologie goed overeen kwam. In het geval van de meldingen kunnen dit kunnen *alle* werknemers zijn (en mogelijk in de toekomst zelfs patiënten zelf). De tekst in meldingen bevat daarom veel meer uiteenlopend en informeler taalgebruik. Dus het aantal uiteenlopende woorden zal voor meldingen aanzienlijk hoger liggen. Ook denken wij dat daarom de kwaliteit van de teksten in de meldingen nooit van het zelfde niveau kunnen zijn als het niveau van de pathologie rapporten.

Het pathologie systeem bouwt vervolgens een multi-dimensionale vector op voor elk goed geclassificeerd rapport. Elk woord is een dimensie en woorden die vaker voorkwamen in een rapport kregen een zwaardere weging in de vector. Woorden die in een groot gedeelte van de vectoren voorkwamen werden gezien als niet erg uniek en werden daarom weggelaten. De tekst in de pathologierapporten was veel omvangrijker. Meldingen zijn in principe ingevulde formulieren en meestal is een groot deel hiervan redelijk gecodificeerd. Hiermee bedoelen we veel ja/nee of meerkeuzevragen in tegenstelling tot vrije tekst. Na het bestuderen van meldingen uit de praktijk merkten wij dat vaak ook korte antwoorden gegeven worden in de vrije tekst. Alsmede dat er soms ook irrelevante zaken in zijn opgenomen in de vrije tekst velden. De kleine hoeveelheid tekst en in vergelijking met het pathologiesysteem kwalitatief lagere tekst maakt het bouwen van goede vectoren bijna onmogelijk.

Conclusie Op basis van deze uitkomsten moet geconcludeerd worden dat een oplossing zoals deze voor de pathologie geschikt was op dit moment niet bruikbaar is op niet-gestandaardiseerde datasets in een VMS. De redenen hiervoor zijn samengevat met name de *kwantiteit* en *kwaliteit* van de vrije tekst. Indien zorginstellingen een of meerdere specialistische tussenpersonen zouden gebruiken die in samenspraak met de melder zorgvuldig een melding invoert als vrije tekst, met gebruik van duidelijke terminologie en een duidelijke omschrijving van het incident. Dan zouden de meldingen in tekst wellicht beter bruikbaar zijn, echter zover hoeven we niet te gaan.

Immers als er dergelijke tussenpersonen zijn aangewezen moeten zij om duidelijke terminologie te kunnen gebruiken enigszins voorbereid worden, dit kost tijd. Deze personen kunnen zich dan net zo goed een analysetechniek en classificatiemodel eigen maken. Een analyse zou een persoon sowieso moeten doen om een goede tekst te schrijven. Een goede uitgebreide tekst schrijven die bruikbaar is voor text-mining

kost waarschijnlijk meer tijd dan het analyseren en classificeren. Daarom is het beter om gelijk over te gaan tot het analyseren van de incidenten en niet meer zo zwaar te hangen aan de vrije tekst.

5.2.2. Latent Semantic Indexing

Een zoekmachine dat gebruik maakt van een LSI database, afhankelijk van het gebruikte algoritme kijkt naar meer karakteristieken van een document. Als we de individuele PRISMA analysen als tekstbestanden zouden zien, met basisoorzaak omschrijvingen uit de analyse als tekst onder elkaar in een document. Dan zouden we met een traditionele zoekmachine bijvoorbeeld met trefwoorden in een aantal documenten kunnen zoeken op "epi". Als een analyse volgens de zoekengine het trefwoord niet bevat wordt deze niet gevonden.

Een typisch voorbeeld hoe dit in zijn werk gaat (gebaseerd op een artikel van NITLE, [42]). Onnodige opmaak wordt weggehaald: verschil tussen hoofd- en kleine- letters, overbodige spaties, kleur, *etc.* Vervolgens worden "stopwoorden" weggefilterd, zoals (in het engels): on, wednesday that, have, to go along with a, *etc.* Met "stemming" wordt verschil tussen enkel- en meerfout weggefilterd (in het engels): happiness en happily naar happi, discouragement naar discourag, *etc.*

Uit deze lijst met woorden worden weggehaald: woorden die slechts in één document voorkomen (deze woorden kunnen niets zeggen over de relaties tussen documenten) en woorden die in alle documenten voorkomen (deze woorden kunnen niet helpen onderscheid te maken tussen de documenten). In een tabel kunnen de documenten als kolommen afgezet worden tegen alle woorden uit de lijst. In de tabel kunnen de woorden onder de documenten gescoord worden d.m.v. nullen en enen (wel- of niet- voorkomend in document) of m.b.v. *term-weighting* bepaalde woorden zwaarder laten meetellen.

Een manier om dit te doen is met behulp van *local-* en *global- term weighting* en *normalization.* Dit is ook hoe *Semantic Engine* (software centraal in Semantic Indexing Project van NITLE) het doet en gebruikt hiervoor het *logarithmic local weighting* algoritme en respectievelijk het *inverse document frequency* algoritme. Met andere woorden, als een woord in een document op een logaritmische as de dubbele afstand zou hebben ten opzichte van een tweede woord, zal deze twee keer zo zwaar worden geteld. Als een woord unieker is in de zin dat het slechts tien keer voorkomt in een document wordt deze zwaarder gewogen dan wanneer deze in honderd documenten zou voorkomen. Het normaliseren houdt in dat omvangrijke documenten (qua aantal karakters) minder zwaar wegen dan kleinere documenten, zodat grote documenten niet alle documenten zullen overschaduwen. De definitieve formule voor de "groter dan nul" waarden in de tabel is: `lokale weging * globale weging * normalisatiefactor`.

Toepassing Ten eerste heb ik een export gedaan van alle basisoorzaakomschrijvingen die bij de analyse van een melding horen in een eigen tekstbestand weggeschreven. Dus analyse1.txt, analyse2.txt, analyse3.txt, ..., elk een aantal basisoorzaak omschrijvingen als plain tekst opgenomen. Elk tekstbestand omvatte dus een aantal regels meestal ongeveer 2 à 5 regels vrije tekst. Daarna de Semantic Engine al deze analysen in een database laten indexeren. Het stemming algoritme heb ik niet aangepast, ook de Engelse stopwoorden niet vervangen door Nederlandse. Dit zorgt voor minder nauwkeurige resultaten.

Als ik de Semantic Engine laat zoeken naar documenten waar "epi" en zijn *similar terms* in voorkomen, worden de volgende termen herkend:

```
(1) Similar terms:  epi, verwerken, grote, gevolgd, doen, uitvoer, kwaliteit, apparatuur, beelden, form
(2) Similar terms:  epi, grote, epi-formulier, bakje, opnamen, tnt, doen, bak, filter, kaart
(3) Similar terms:  epi, gevolgd, filter, opname, afwijkende, auto, apparatuur, epi-formulier, protokol, epidos
```

Voor het vinden van deze similar terms zijn met de voorbeeld tools die met de sourcecode meegeleverd waren gebruikt. De tool is drie keer uitgevoerd, vandaar de drie verschillende resultaten. In de im-

plementatie van deze meegeleverde tools wordt een "random walk" algoritme toegepast om irrelevante indexing data weg te filteren voor het genereren van de subgraven. "We have found this to dramatically increase the relevance of returned results, while also dramatically decreasing the search time." [39] [10].

Wanneer door analisten over Epid gesproken werd ging het ook vaak over problemen met de apparatuur, dat de contrast/kwaliteit van de beelden niet goed waren, *etc.* Hier zijn destijds veel meldingen van gemaakt en het goede nieuws is, dat toepassing van LSI ervoor kan zorgen dat deze trefwoorden met elkaar geassocieerd worden (epi, kwaliteit, apparatuur, beelden, filter, *etc.*), ondanks de aard van de input (zie subsectie 5.2.1).

Conclusie De tool kan clusteren op basis van de gelegde associaties tussen de begrippen en dat is gedaan op drie verschillend ingerichte databases. De resultaten hiervan zijn deels opgenomen in appendix F. In één oogopslag kan het lijken of de clusters verschillen, maar het kan zijn dat de trefwoorden anders gekozen zijn terwijl ze wel semantisch dichtbij elkaar liggen. Om dit te verifiëren zullen in de toekomst naast de standaard meegeleverde voorbeeld tools queries gedaan moeten worden op de databases.

Latent Semantic Indexing legt misschien zeer fragiele semantische associaties tussen de termen omdat de tekst zo "dun" is. Maar hij legt ze wel, en als we van te voren afspreken dat dit geen exact instrument is kan het wel een interessante alternatieve kijk op de informatie verschaffen. Immers het geheel dat uit een heleboel dunne tekst bestaat is toch 267KB aan *plain*tekst aan oorzaakomschrijvingen. Semantische associaties tussen begrippen binnen deze hele tekstset kunnen nog steeds wel of niet waardevol zijn. Daarnaast is ook nog een hoop informatie vastgelegd dat voor mijn (hoofd)onderzoek niet direct relevant was op e.g. incidentniveau en oorzaakomschrijvingen in de analyseboom van overige (niet-basis) oorzaken leaves. Dit erbij betrekken en een aanpassing voor nederlandse stemming is iets dat in de toekomst uitgeprobeerd kan worden. ■

[10] Alle subgraph policies zijn configureerbaar, ik heb de standaard instellingen gebruikt.

6 Benchmarking PRISMA gegevensbron

In dit hoofdstuk wordt op twee subonderzoeksvragen antwoord gegeven, ten eerste Sub-3.1 (opgedeeld):

- Hoe kunnen we de *huidige werkwijze* van de PRISMA analist automatiseren en ondersteunen in de data-analyse (drilldown deductieproces) en benchmarking?
- Wat is hiervan de toegevoegde waarde?

De huidige werkwijze van de analist is te omschrijven als het nastreven van een ideale situatie, *e.g.* alle typen fouten dienen evenveel voor te komen. Op het moment dat *e.g.*, een technische basisoorzaak (Technical/Design, TD) veel hoger blijkt te zijn, wordt door middel van een drilldown proces zo specifiek mogelijk achterhaald waar deze hoge meting vandaan komt (hiervoor worden dus de contextvariabelen gebruikt). Aan de hand van de PRISMA methodiek en de contextvariabelen wordt een juiste verbetermaatregel opgesteld, die ervoor moet zorgen dat de hoge tellingen van de TD classificatie weer afneemt, en de metingen meer neigen naar de ideale situatie.

Het analyse- en benchmarkingproces kan op een waardevolle manier ondersteund worden door een aantal praktische implementaties, zoals de ContextCompareView. Dit is een visuele weergave van de contextvariabelen die ervoor zorgt dat de analist ten eerste niet op doodlopende sporen terecht komt, en ten tweede wordt in één oogopslag visueel weergegeven hoe de spreiding is van de metingen over de contextvariabelen (*i.e.*, in welke processen komt basisoorzaak Technical/Design (TD) voornamelijk voor). Deze visualisatie is onder andere geïmplementeerd in het prototype en heeft toegevoegde waarde omdat een hoop stappen in het analyseproces nu sneller en makkelijker worden.

Ten tweede subonderzoeksvraag Sub-3.2 (opgedeeld):

- Hoe kunnen als aanvulling op de Sub-3.1 de uitkomsten van Sub-1 (kansberekening) en Sub-2 (data-mining technieken) toegepast worden in het deductieproces en benchmarking?
- Wat is hiervan de toegevoegde waarde?

Het benchmarkproces kan ondersteund worden met name op het gebied van efficientie. De analist kan twee instituten A en B naast elkaar zetten op allerlei manieren. Daar komen verschillen uit en voordat de analist deze verschillen gebruikt om beslissingen te nemen voor procesaanpassingen, kan deze door middel van kansberekening nog beter geïnformeerd worden. Door procesaanpassingen te maken waarvan de waarschijnlijkheid het hoogst is dat de gemeten verschillen er daadwerkelijk zijn tussen de instituten (en niet door pure willekeurigheid van de steekproef zijn ontstaan), handelt de analist efficienter.

Een techniek die zich leent om dezelfde kansberekening toe te passen voor procescontrole en benchmarking is de Control Chart. Gevisualiseerd kan worden (met een bepaalde zekerheid) in welke mate een proces in controle is, omdat van elke meting te zien is in hoeverre deze volgens verwachting is. In het kader van benchmarking kunnen metingen van een individueel instituut A tegen het landelijke gemiddelde gehouden worden. Indien hier blijkt dat bepaalde processen van A afwijken van het landelijke gemiddelde, kunnen hier verbetermaatregelen op los worden gelaten.

Toegevoegde waarde van clustering op basis van K-Means, Latent Semantic Indexing en op compressie is er op dit moment nog niet met betrekking tot benchmarking (met uitzondering van N-Pairs, dat

alvast besproken is in hoofdstuk 5). Hiervoor is nodig dat de contextvariabelen beter gedefinieerd en gehanteerd worden. Door het naast redeneren ook de gegevens te visualiseren, wordt aangetoond dat het nog te vroeg is om over concrete toepassingen te spreken.

Wat benchmarking precies is en hoe het een waardevolle rol kan innemen in een organisatie wordt kort besproken in sectie 6.1. De huidige werkwijze en de PRISMAView en ContextCompareView in het prototype worden in sectie 6.2 besproken. In sectie 6.3 wordt ingegaan op de toepassingen voor kansberekeningen, met name groepenvergelijking op basis van Chi-kwadraat en de Control Charts. Tot slot wordt kort beschouwd wat mogelijke toepassingen zijn van data-mining en waarom ze nog niet werken in sectie 6.4.

6.1. Benchmarking in het algemeen

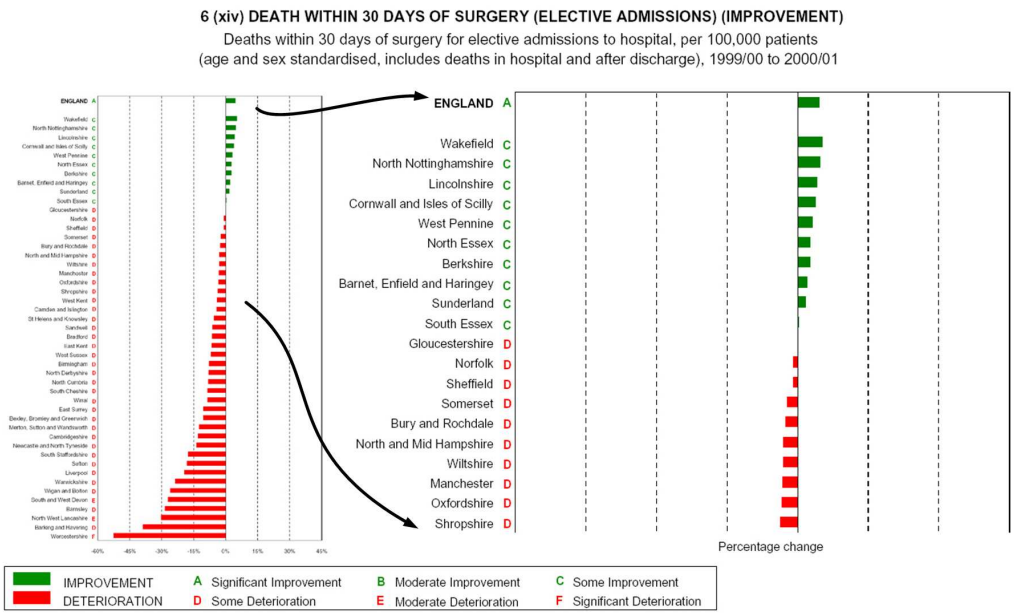
Het RT-initiatief is een samenwerking en geen concurrentiestrijd, daardoor valt het type benchmarking onder *competitive* benchmarking waar *niet*-interne processen vergeleken worden (waarin wel soortgelijke behandelingen worden uitgevoerd). In de medische wetenschappelijke literatuur [24] [8] is het gebruikelijk om onderscheid te maken in vier typen benchmarking: "With internal benchmarking, functions within an organization are compared with each other. Competitive benchmarking partners do business in the same market and provide a direct comparison of products or services. Functional and generic benchmarking are performed with organizations which may have a specific similar function, such as payroll or purchasing, but which otherwise are in a different business." [8].

Bensor *et.al.* gegeven aan in [8] hoe belangrijk het vervolgproces is: "Benchmarking must be a team process because the outcome will involve changing current practices, with effects felt throughout the organization. The team should include members who have subject knowledge; communications and computer proficiency; skills as facilitators and outside contacts; and sponsorship of senior management." Dit geldt ook voor benchmarking in de Radiotherapie, vaak zullen de cijfers die voortkomen uit de benchmarking niet direct leiden tot verbetermaatregelen, maar wel tot ontmoetingen tussen analisten van twee instituten die overgaan tot nadere procesvergelijking.

De conclusie van een onderzoek naar twee internal benchmarking case studies [27] was: "Two key factors of a successful benchmarking initiative are highlighted: management commitment and involvement, and the need to understand internal practices and processes before benchmarking.". De PRISMA RT expert-groep (MAASTRO, CZE, ZRTI, *et.al.*) realiseerde zich dit ook, het tweede punt is de rede geweest dat onderscheid gemaakt wordt tussen "interne" en "externe" contextvariabelen. Interne contextvariabelen zoals concrete werkeenheden, zijn alleen voor de instelling zelf beschikbaar om verkeerde hantering door andere instellingen te voorkomen.

Nu volgen drie paragraven waarin kort drie verschillende aanpakken besproken worden die in benchmarking mogelijk zijn.

Performance indicatoren Het visualiseren van de performance in de zin van verbetering en verslechtering, zoals in figuur 6.1, is in het geval van de Radiotherapie zinnig om te doen op eenzelfde soort manier. Alleen de performance indicatoren zijn dan op basis van classificaties, *i.e.*, 2% tot 14% T-EX in combinatie met avonddienst. In de zorgsector wordt op vele vlakken met dergelijke indicatoren gewerkt, voorbeelden: NHS performance Indicatoren, de prestatie-indicatoren van de Inspectie voor de Gezondheidszorg [36]. Meer is bijvoorbeeld te vinden in [11] over het gebruik van prestatie indicatoren, potentieel gebruik en potentieel *verkeerd* gebruik in Amerika.



Figuur 6.1: Voorbeeld gebruik NHS ([17, p. 149]), performance ziekenhuizen.

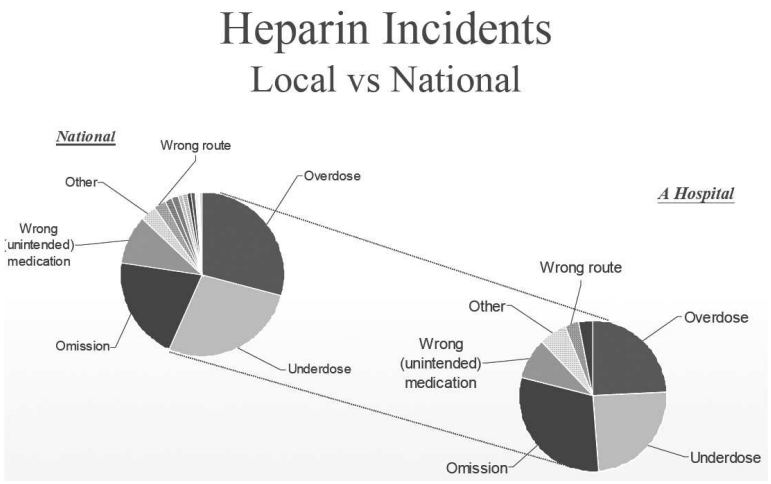
Best-practice benchmarking De best-practice op een bepaalde "performance indicator"—T-EX en avonddienst—kan zijn het instituut die daar relatief gezien het minst van in aantal of percentage heeft. Of het instituut dat deze performance indicator in het afgelopen kwartaal of half jaar heeft weten terug te dringen. Overige instituten kunnen overgaan tot procesvergelijking met deze best-practice of nagaan welke verbetermaatregelen deze best-practice onlangs gekomen heeft. Hierin is direct koppelen van verbetermaatregelen in het benchmarksysteem niet mogelijk omdat deze ziekenhuisbreed worden bijgehouden. Naast een specifiek instituut als best-practice kan ook het landelijk gemiddelde als best-practice gehanteerd worden, zoals te zien in figuur 6.2.

Benchmarking en monitoring met procesvariatie Variatie van een proces kan gemeten worden om een bepaalde centrale middenlijn. Hiervoor kan bijvoorbeeld het gemiddelde gebruikt worden, maar in het kader van benchmarking ook gemiddelden van andere afdelingen, werkeenheden, *etc.* In het prototype zullen Control Charts gebruikt gaan worden om de schommeling te visualiseren en te kunnen beoordelen op urgentie (zie 6.3). Daarnaast dienen ze oorspronkelijk als controle mechanisme en daarvoor kunnen ze uiteindelijk ook gebruikt gaan worden. Toekomstige uitbreidingen: vastzetten van grenzen, het verscherpen van grenzen en het veranderen van de grenzen op een specifiek moment, *i.e.*, een interventie.

6.2. Benchmarking en de huidige werkwijze

Benchmarkmetrieken kunnen volgens [8] zijn: productiviteit, kwaliteit, tijd en kosten-gerelateerd. Wij richten ons dus op *kwaliteit*, concrete voorbeelden van hoe wij zullen benchmarken:

- Twee soortgelijke processen, werkeenheden binnen een instituut onderling met elkaar vergelijken.
- Twee soortgelijke processen, werkeenheden van een instituut of het instituut in zijn geheel met een tweede vergelijkbare partij. Deze partij kan zijn de best-practice, het gemiddelde van alle instituten of een fictief "ideaal instituut" dat gebaseerd is op streefwaarden of andere input/prognoses.



Figuur 6.2: Voorbeeld gebruik APSF ([5, p. 111]), benchmarking met landelijk gemiddelden.

Drill-down functionaliteit in prototype In het prototype staat het PRISMA profiel centraal. Op de X-as kunnen objecten als groepen uiteengezet worden: instanties, werkeenheden of processen. Op deze X-as zijn voor elk object de classificaties uitgezet met over de Y-as de absolute waarden of percentages (instelbaar). Alles in dit scherm is volgens het "drill-down" principe: de perioden (van/tot), contextvariabelen en keuzen en de classificaties. De classificaties kunnen ook geselecteerd worden en dan verschijnt een tweede grafiek waarin de classificatie in de tijd is te bezichtigen (voor dezelfde objecten). Dit is de primaire view in het prototype: de PRISMAView (screenshot figuur B.4).

Naarmate in de drill-down deductie techniek een scherpere selectie—of "set van basisoorzaken"—wordt gemaakt de gegevensbron, dunt deze snel uit. Na het opleggen van een aantal beperkingen in de contextvariabelen, *i.e.*, "alleen de TD classificaties waar sprake was van een avonddienst en personeel dat minder dan één jaar in dienst is", blijven er vaak weinig tot geen basisoorzaken over.

Aanpassing van het prototype Eén van de eerste toevoegingen aan het prototype is geweest de tabel in de ContextCompareView (screenshot figuur B.5). Daar staat per contextvariabel hoe vaak elke keuze is voorgekomen in de selectie. Door de PRISMAView en de ContextCompareView af te wisselen De ContextCompare weergave is een tabel die voornamelijk verticaal veel ruimte inneemt en leent zich daarom prima voor het naast elkaar zetten van meerdere instituten tegelijk. worden in het analyseproces doodlopende sporen vermeden omdat ten alle tijde zichtbaar is of er metingen aanwezig zijn. Daarnaast geeft de tabel zelf informatie dat al kan leiden tot inzicht in het probleem omdat de spreiding van contextvariabelen op een overzichtelijke manier gepresenteerd wordt. Op basis van een tweede toevoeging, de TrechterView [1], kan in de onderliggende teksten (de omschrijvingen van de basisoorzaken, afkomstig uit de PRISMA analyseboom [2]), met hulpmiddelen gezocht worden naar patronen die ook meer inzicht kunnen geven in de oorzaak.

Praktijk Volgens een bepaald ideaalbeeld (*i.e.*, alle bars hetzelfde percentage, of een gezamenlijk profiel, ...) wordt gestuurd op het PRISMA profiel van het RT instituut voor afdelingen. Deze profielen kunnen naast elkaar gelegd worden indien nodig en de analist kan op zoek gaan naar verschillen in classificaties, contextvariabelen, omschrijvingen, en zo hopelijk tot een potentiële verklaring komen. Meestal is het no-

[1] Was beschikbaar gedurende het (hoofd)onderzoek, is nog in ontwikkeling en wordt in deze scriptie niet op ingegaan.
[2] In de toekomst misschien nog in andere velden zoals Incident omschrijvingen, *etc.*, niets is uitgesloten.

dig om deze te verifiëren (*i.e.*, op basis van ervaring of navragen). In combinatie met de actie/interventie matrix kan op een redelijk verantwoorde manier vervolg gegeven worden aan de bevindingen.

Een tweede voorbeeld, een onverwachte verhouding in de ContextCompare View kan aanleiding zijn voor het toevoegen van deze uitschietende keuze aan de selectie/drilldown, *i.e.*, "apparatuur ouder dan 5 jaar". Om vervolgens voor deze selectie te kijken hoe de contextvariabelen zich vervolgens weer verhouden, om tot de conclusie te komen dat "apparatuur ouder dan 5 jaar" in 90% van de gevallen voorkomt in combinatie met "speciale omstandigheid: avonddienst". In dit fictieve voorbeeld zou dat kunnen betekenen dat de betreffende apparatuur rond bepaalde tijdstippen wellicht niet goed functioneert.

6.3. Benchmarking en kansberekening

Heeft statistisch gezien de genomen verbetermaatregel geholpen *e.g.*, in hoeverre was de daling na de interventie te verwachten? Dit is te visualiseren door middel van een op kansberekening gebaseerde Control Chart (afkomstig uit de procescontrole, zie subsectie 6.3.1). Hoe groot is de kans dat het hoogste kwartaalcijfer aan toeval toe te schrijven is? Hoeveel (bijna-) incidenten kunnen we verwachten in het volgende jaar (*e.g.*, 1000) en wat is daarvan de foutmarge (*e.g.*, ± 50 incidenten[3])? Deze vragen zijn te beantwoorden op basis van de Poisson verdeling (subsectie 6.3.2).

Hoe groot is de kans dat er sprake is van een *te verwachten* afwijking op een bepaald gebied (*e.g.*, T-EX + avonddienst), tussen instituut A en en het landelijke gemiddelde of landelijke best-practice? Deze vragen zijn ook te beantwoorden door gebruik te maken van de Poisson verdeling. Literatuur over dit onderwerp is gevonden[4], echter is niet bestudeerd in- of betrokken bij dit onderzoek. Gekozen is om de Pearson Chi-kwadraat test voor deze uit te proberen (subsectie 6.3.3).

Definitie "significantie" volgens de Van Dale is de betekenis van significant (bijvoeglijk naamwoord; significanter, significantst; significantie): (i) veelbetekenend, (ii) statistisch verantwoorde conclusies toelatend.

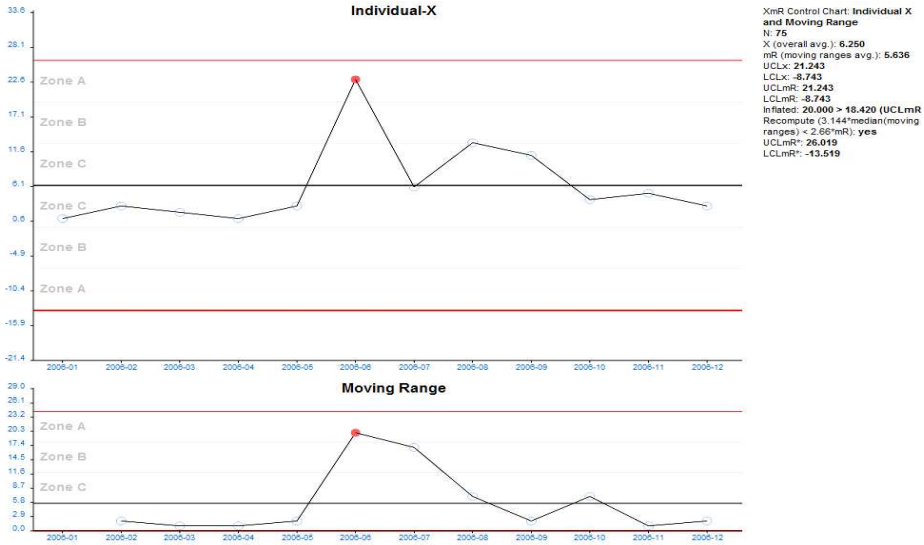
6.3.1. Procescontrole met Control Charts

De Control Chart is een statistisch hulpmiddel om binnen *proces variatie* de *speciale oorzaken (assignable causes)* te onderscheiden van de *procesinherente oorzaken (chance causes)* ([35, samenvatting p.95], [19]). Een proces is *in control* wanneer er alleen sprake is van procesinherente variatie. In de aard van een specifiek proces zit statistisch gezien een willekeurige schommeling (rondom een gemiddelde) die natuurlijk en procesinherent is. Een speciale oorzaak op de variatie in een proces kan zijn een verhuizing, een nieuw team of een overstap naar nieuwe software.

De opbouw is, (i) de metingen in de tijd, (ii) een middenlijn met het gemiddelde en (iii) *control limits* of "controle grenzen" [6]. Deze grenzen omvatten de variatie en worden afhankelijk van het type control chart berekend met bepaalde formules. Onder andere metingen buiten deze grenzen zijn dermate onwaarschijnlijk dat er waarschijnlijk sprake moet zijn van speciale oorzaken (andere waarnemingen kunnen net zo onwaarschijnlijk zijn *i.e.*, acht metingen aan dezelfde kant van de middenlijn, *etc.*). In de grensberekening worden constanten gebruikt van Shewhart (te vinden in o.a. [23, 6]). Deze constanten zijn vooraf uitgerekende formules die het mogelijk maken om onder- en bovengrenzen van (meestal) ± 3σ (sigma) rondom de middenlijn zonder te veel moeite berekenen. In de control chart omvat het gebied binnen de grenzen 6σ 99.7%.

[3] *i.e.*, de werkelijke waarde ligt met 95% zekerheid 50 lager of hoger dan 1000.
[4] *i.e.*, [2], [16]

Verschillende control charts zijn geschikt voor bepaalde typen data [5]. In dit onderzoek is de XmR (*Individual-X and Moving Range*) chart gebruikt (zie figuur 6.3) voor het berekenen van de grenzen, indien de grenzen "te ruim" kunnen en wordt een correctie toegepast [31]. Dit is een control chart waar één "groep" in de tijd uiteengezet wordt, *i.e.*, aantallen per maand, alsmede de "moving range", aantallen per twee maanden (huidige en vorige). Deze laatste kan gebruikt worden indien zeer weinig metingen beschikbaar zijn, iets dat voor kan komen in het drilldownproces. Alle regels met betrekking tot het interpreteren van de XmR chart zijn te vinden in appendix E.



Figuur 6.3: XmR MAASTRO 2006 Epid, organisatorische faalwijzen. Niet genormaliseerd.

Toepassing Om te kunnen compenseren voor oorzaken die door alle metingen (technische, organisatorische, ...) is de XmR chart voorzien van een normalisatieoptie in het prototype (voor details zie appendix E). Deze normalisatieoptie houdt in een "relatieve" correctie van een bepaald type faalwijze op basis van de verhouding die deze heeft ten opzichte van *alle* type faalwijzen, *i.e.* technisch falen ten opzichte van alle falen. Deze normalisatie is toegevoegd zodat een algemene stijging in de meldfrequentie in het gehele instituut gecompenseerd wordt. Deze compensatie werkt alleen als: de *speciale oorzaak* die we willen detecteren optreedt in een *subset* van de mogelijke typen faalwijzen, en dus niet in alle. Als een speciale oorzaak invloed heeft op alle typen faalwijzen wordt het effect namelijk gemaskeerd (dan corrigeert het zichzelf). Het lastige is dit op voorhand niet bekend is voor de analist, vandaar dat de normalisatie optioneel is aangeboden in het prototype.

Conclusie Na de implementatie van de Control Chart en toepassing op de processen binnen de MAASTRO was de meest opvallende uitschieter in het Epid proces, te zien in figuur 6.3. Na normalisatie was hier als enige sprake van een uitschieter zelfs buiten de 6 sigma grenzen. Omdat dit na overleg ook precies de afdeling was waar ze in dat jaar de meeste problemen hebben gehad vanwege de verhuizing lijkt het erop dat het in staat is in ieder geval de grove uitschieters te herkennen. Een telling van aantal geregistreerde incidenten per proces bijvoorbeeld zou niet genoeg zijn geweest omdat Linac bediening het grootste proces is en niet een dergelijke schommeling laat zien in de Control Chart. In het valideren

[5] Voor een lijst van veelgebruikte control charts en beslisbomen zie figuur E.1 (afkomstig uit [23, p. 419])

van de prototype, zoals omschreven in sectie 3.4.1, worden meer analysen uitgevoerd met de Control Charts, in hoofdstuk 7 staan de resultaten.

6.3.2. Kansberekening met Poisson verdeling

Vanwege de additiviteit van de Poisson verdeling is het mogelijk de kans te berekenen op *e.g.*, 100 T-EX per kwartaal, ondanks dat de Poisson distributie opgebouwd is uit metingen per week. Dat de poisson op deze manier additief is staat beschreven in [12, p. 6] met verwijzingen naar details. In de Poisson zijn de lambda (gemiddelde) en variantie hetzelfde, daarom kan gezegd worden dat de lambda voor een kwartaal bestaat uit 13 weken. Concreet houdt dit in, dat de kans op 100 T-EX in een kwartaal indien de lambda per week 7 is, de lambda vermenigvuldigd kan worden met 13 weken en in de berekeningen daarom een lambda van 91 gebruikt moet worden.

Indien de analist een efficiënte verbetermaatregel wilt nemen kan deze de classificatie met de meeste metingen uitkiezen en deze in de tijd uiteenzetten per kwartaal. Het doel hiervan kan zijn om te kijken of er sprake is van een specifieke periode waar een speciale oorzaak heeft plaatsgevonden die heeft geleid tot deze omvangrijke classificatie. Hiervoor kan de poisson verdeling geïntegreerd worden in de software zoals in figuur 6.4, door de waarschijnlijkheid van de metingen weer te geven in de bars. De waarschijnlijkheid kan in percentages worden weergegeven of zoals in het figuur door middel van p-waarden (gebaseerd op *e.g.*, 95% zekerheid). De goodness-of-fit functie kan dienen als een preconditie voor het overgaan tot dit soort significantieberekeningen.

Figuur 6.4: Significantie toevoegingen PRISMA classificatie in de tijd grafiek (voorbeeld).

6.3.3. Pearson Chi-kwadraat voor vergelijken groepen

De analist kan zelf beoordelen of een groep wel of niet serieus afwijkt van een andere groep, maar omdat we de theoretische verdeling hebben kan dit ook met een formule. Volgens de literatuur is de chi-kwadraat test geschikt, daarom gaan we deze uitproberen op de MAASTRO gegevens. De chi-kwadraat toets is te gebruiken om te bepalen of de *waargenomen* metingen significant afwijken van de *verwachte* metingen. De formule voor chi-kwadraat (χ^2) [6]:

$$\chi^2_{n-1} = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$$

[6]Bron chi-square goodness of fit test: `http://www.stat.yale.edu/Courses/1997-98/101/chigf.htm`.

Hierin staat O_i voor een waargenomen- en E_i voor een verwachte- meting ("observed" en "expected"), n is het aantal metingen. Vervolgens kan met dit statistiek en de *cumulatieve distributie functie* [40] de p-waarde uitgerekend worden (om de significantie in uit te drukken van de afwijking)[7]. Lage p-waarden geven aan dat er bewijs is tegen de nulhypothese die luidt dat er géén verschil is tussen de groepen. Vanaf 0.05 wordt het significant en nog sterker significant richting de nul.

Indien we met de test de significantiecijfers tussen een aantal uiteenlopende afdelingen (A, B, C en D) en het totale instituut uitrekenen. Geeft de test oordelen die overeenkomen met de verwachtingen, zie tabel 6.5. De observed metingen zijn de kwartalen die bij de afdelingen horen, *i.e.*, bij afdeling B zijn dat 6, 32, 2, *etc.* Bijbehorende expected values zijn *niet* de totalen van de organisatie—67, 151, 82, ...—maar omgerekend naar de omvang van de afdeling (totaal afdeling B / totaal alle afdelingen), dus: (67 * (56/787)=)4.77, 10.74, 5.83, *etc.*

Volgens SPSS 16.0 documentatie is een vereiste voor het gebruik van de test dat: "The expected frequencies for each category should be at least 1. No more than 20% of the categories should have expected frequencies of less than 5.". Deze eis is heel belangrijk voor in het prototype omdat het zoals in het voorbeeld te zien is dat de eis niet door alle afdelingen gehaald wordt. In dat geval kan er niks gezegd worden over significant of niet. Meerdere afdelingen voldoen wel aan de eis, zoals ook alle werkeenheden en dus ook instituten.

	Som	p-waarden	Metingen (n=11 kwartalen)										
Alle afdelingen	787		67	151	82	65	58	68	89	59	83	41	24
Afdeling A	5	0.31547	1	1	0	0	2	1	0	0	0	0	0
Afdeling B	56	**0.00000**	6	32	2	0	0	0	1	5	8	2	0
Afdeling C	135	**0.00028**	16	8	16	5	5	14	27	11	17	9	7
Afdeling D	129	**0.08008**	10	31	16	13	8	10	7	4	21	8	1

Figuur 6.5: Chi-kwadraat toegepast op voorbeeld groepen.

Observaties De resultaten uit het voorbeeld demonstreren p-waarden die volgens verwachting zijn. Het proces A schommelt met een standaarddeviatie van 0.68755 om het gemiddelde 0.45455 heen, er is maar één meting van de elf die boven de standaarddeviatie uitkomt. In tegenstelling tot proces B, hier is het gemiddelde 5.09091, de SD 9.34296 en als naar de invoer gekeken wordt valt gelijk de uitschieter van 32 wel erg op ten opzichte van de overige metingen. Of hier echt wat aan de hand is staat niet vast, maar statistisch gezien is deze uitschieter wel significant. Het afzetten van de processen tegen de gehele instelling als "gemiddeld proces" geeft p-waarden die met de bij dit voorbeeld horende verwachtingen overeenkomen.

Conclusie De chi-kwadraat toets is bruikbaar om significantie in mate van afwijking te bepalen tussen groepen. Gunstige bijkomstigheid van de chi-kwadraat toets is dat een proces, zoals proces A met zeer weinig metingen in deze selectie, niet (zomaar) kan zorgen voor een significante uitkomst.

[7] bron: `http://en.wikipedia.org/wiki/Chi_square`:
$F(x;k) = \frac{\gamma(k/2, x/2)}{\Gamma(k/2)} = P(k/2, x/2)$
"where $\gamma(k,z)$ is the "lower incomplete Gamma function" and $P(k,z)$ is the "regularized Gamma function".

6.4. Benchmarking en data-mining

Benchmarking in combinatie met data-mining kan verschillende vormen aannemen. (i) De tot nu toe omschreven manier *i.e.*, een werkeenheid ten opzichte van het landelijk gemiddelde (dit kan bijvoorbeeld ook nog voor combinaties van contextvariabelen). (ii) Zoeken naar afwijkende uitzonderingsgroepen *i.e.*, een bepaalde groep (cluster gebaseerd op K-Means, LSI of CompLearn) komt overal voor behalve bij een enkele instelling (of visa versa). Text-analyse kan in deze bepalen waar de uitzondering (instituut) zit welke bepaalde concepten (of clusters van concepten) niet heeft die anderen juist wel hebben. (iii) Best-practice selectie op basis van tot nu toe beschreven technieken *i.e.* welk instituut heeft een soortgelijk of juist tegenovergesteld profiel? welk instituut presteert het beste? Hier kan uit de data-mining de Kolmogorov Complexiteit benadering ook nog van toepassing zijn, zie paragraaf 5.1.3.

Een niet-benchmarking toepassing is patroonherkenning als "alert" mechanisme. De groepen kunnen in een tijdlijn uiteengezet worden en dan kan gedetecteerd worden of een bepaalde groep ergens ineens weg valt —wellicht omdat de onderliggende oorzaak is opgelost—of een bepaalde nieuwe groep zich juist vormt binnen één of meerdere instituten. Beide gevallen zijn interessant om te monitoren.

De data-mining toepassingen zullen gevisualiseerd moeten worden indien ze in de toekomst—na verder onderzoek—geïmplementeerd worden in het benchmarkpunt. Dit brengt een aantal extra requirements met zich mee, onder andere drill-down functies. De eerder omschreven drill-down functionaliteit maar ook een drill-down in de zin van een filter dat bepaalde irrelevante groepen kan uitschakelen (groepen die volgens de analist logisch zijn en niets zeggend zijn wil deze niet constant in zijn rapportages terug zien). Verder zijn er een hoop zaken waar rekening mee gehouden en grondig naar gekeken zal moeten worden (*e.g.*, outliers detectie, wat daarmee te doen, thresholds om te bepalen welke clusters "voldoende" zijn op wat voor manier dan ook, ...). De juiste waardes zullen gevonden moeten worden voor dit type data en om *e.g.* een alerting systeem niet op overbodige momenten af te laten gaan. Hier valt dermate veel te onderzoeken dat in dit onderzoek geen uitgebreide conclusies gemaakt kunnen worden. ■

7 Resultaten

De resultaten van de prototype evaluatie op basis van het script zijn opgenomen in sectie 7.1. Ondanks de conclusie dat K-Means nu geen zin heeft worden de clusters nader onderzocht in 7.2. Verder volgen samenvattingen van de eerdere resultaten voor text-mining analyse in 7.3, en kansberekening in 7.4.

7.1. Prototype evaluatie

Uitvoering De in subsectie 3.4.1 omschreven evaluatie heeft 12 december 2007 plaatsgevonden, heeft ongeveer vijf uur in beslag genomen. De acht basisanalysen uit tabel 7.1 zijn hier uitgevoerd. De analisten waren Jørgen van den Bogaard en Pascale Simmons, mijn functie was notuleren en eventuele ondersteuning in de bediening[1]. Tijdens het analyseren kwamen een aantal requirements die al in de wachtrij stonden opnieuw naar boven en een aantal nieuwe, deze zijn geregistreerd maar niet meer opgenomen in dit verslag. Het belangrijkste deel[2] van de genotuleerde observaties in de gemaakte analysen is letterlijk en gestructureerd opgenomen in tabellen te vinden in appendix C.

Gedurende het uitvoeren van dit experiment op de computer wordt alles op het scherm opgenomen. Per gevonden observatie wordt het tijdstip genoteerd zodat achteraf de volledige flow altijd weer te bestuderen is. Tijdens het experiment zal ik requirements noteren voor het vervolg van het project *na* dit afstudeeronderzoek.

Analyse	Selectie	Classificatie
Analyse #1	Werkeenheden 1, 2, 3	OP (Organisational Protocols)
Analyse #2	Proces Epid	OP (Organisational Protocols) [vervolganalyse]
Analyse #3	Proces Epid	TD (Technical Design) [vervolganalyse]
Analyse #4	Werkeenheden 1, 2, 3	OM (Organisational Management priorities)
Analyse #5	Werkeenheden 1, 2, 3	OC (Organisational Cultures)
Analyse #6	Werkeenheden 1, 2, 3	TD (Technical Design)
Analyse #7	Werkeenheden 1, 2, 3	HRV (Human Verification) (CC's)
Analyse #8	Werkeenheden 1, 2, 3	OP (Organisational Protocols) (CC's)

Tabel 7.1: Uitgevoerde analysen MAASTRO op basis van draaiboek, d.d. 12-dec.2007.

Resultaten In totaal zijn uit de 8 vertrekpunten 13 observaties genoteerd, zijn tijdens de analyse en drill-down 13 hypothesen geformuleerd (niet elke observatie leidde tot een hypothese en sommige tot meerdere). Hiervan konden 10 hypothesen gefalsificeerd worden en 3 niet. Van deze drie is één keer een vervolghypothese geformuleerd en één keer was het nodig voor de analist om over te gaan tot nader onderzoek.

[1] *i.e.*, de eerder aangehaalde TrechterView is nog niet via het prototype aanroepbaar door de analisten, dit kon ik af en toe handmatig doen, zodat we er toch gebruik van konden maken om e.e.a. te kunnen verifiëren.

[2] De volgende zaken; requirements/feature-requests, feedback, uitgebreide onderliggende redenaties/discussie, zijn weggelaten in deze tabellen.

De analisten maakten in alle gevallen van oorzakenanalyse gebruik van hun domeinkennis en persoonlijke herinneringen (de analisten zijn zelf ook actief op de werkvloer). In overige gevallen was het bijna altijd nodig om de basisoorzaakomschrijvingen erbij te pakken middels de TrechterView om bepaalde zekerheid te geven. Ze hadden ook om andere redenen al kennis over de gegevensbron omdat meerdere TU/e afstudeerders (Jeroen Rutteman en Jasper Weterings) bepaalde aspecten van de gegevensbron ook aan het analyseren waren (*e.g.*, om te onderzoeken hoe meldgedrag invloeden heeft op de gegevensbron).

In dat jaar waren ook een hoop (gefaseerde) verhuizingen van Heerlen naar Maastricht, en omdat wij ons op de efficiëntie richten (de "grove" problemen) waren veel observaties te herleiden tot de verhuizing. Samengevat de resultaten per view in het prototype in tabel 7.2.

View in prototype	Resultaat
PrismaView	Visualisatie is goed, hoge scores per kwartaal zijn alleen lastig te interpreteren voor een analist (significantie).
ContextCompareView	Visualisatie geeft veel extra informatie en zorgt ervoor dat doodlopende einden vermeden worden.
ControlChartView	Van alle observaties correspondeerde een gebeurtenis dat betrekking had op de verhuizing, dus deze activiteiten worden goed gevisualiseerd. Is nog niet in benchmarking verband gebruikt.
K-MeansView	De analist kan de uitkomst(en) van de clustering niet interpreteren. Oorzaak is de kwaliteit van de contextvariabelen.
TrechterView	In prototype deels geïmplementeerd, requirements zijn opgesteld.

Tabel 7.2: Resultaten per view in prototype samengevat.

7.2. Clustering analyse

Om toch wat meer te weten te kunnen komen over de kwaliteit van de clusters en niet alleen afhankelijk te zijn van de interpretatie door de analisten, zijn de in subsectie 5.1.1 besproken visualisaties ten behoeve van het beoordelen van de clusters uitgevoerd. Zie figuren 7.1, 7.3 en 7.5, en de resultaten samengevat in tabel 7.3. De resultaten met betrekking de verschillende clustering technieken uit sectie 5.1 zijn samengevat in tabel 7.4.

Toelichting 3D scatterplot In de clustering van de gegevens worden alle contextvariabelen als dimensie gebruikt. In de visualisatie zijn ze gegroepeerd op basis van de drie assen: menselijk en overige (X-as), technisch (Y-as) en organisatorisch (Z-as).

Op het niveau van alle basisoorzaken (fig. 7.1) zijn er nauwelijks clusters te herkennen, zeven clusters zijn gekleurd echter blijft de vorm één grote "blob" van punten in de grafiek. Op het niveau van individuele basisoorzaken *e.g.*, Technisch Design (TD), zijn vaak wel grove clusters te herkennen. Deze clusters zijn echter vooral in de visualisatie goed te zien omdat de contextvariabelen gegroepeerd worden, en dus eerder een bepaalde richting in gestuurd worden. Drie technische contextvariabelen worden onder de groep technisch bij elkaar opgeteld (techniek wordt één dimensie), waar normaal in de clustering de drie contextvariabelen drie aparte dimensies zouden zijn voor de basisoorzaak vector. Een visueel zichtbare groep van basisoorzaken hoeft niet in één cluster ingedeeld te zijn omdat op het niveau van *alle* dimensies (niet geaggregeerd) om deze rede door andere basisoorzaken heenloopt. Om dit te verifiëren is in het clusterproces geëxperimenteerd door alvast de aggregatie toe te passen en daarna de clusters te vormen, het resultaat hiervan is fig. 7.3. Drie contextvariabelen gebruiken in plaats van aggregatie is bijvoorbeeld niet mogelijk omdat niet elke basisoorzaak dezelfde contextvariabelen heeft.

Clustering op basis van de drie assen (dus de samengenomen contextvariabelen) zorgt voor duidelijker clusters maar is minder bruikbaar. De informatie die uit de visualisatie te halen is wordt gelijk een stuk algemener, vanwege de groepering is een bepaalde plaatsing op een plek in de ruimte van de scatterplot op verschillende manieren mogelijk. Bepaalde overeenkomsten op basis van specifieke factoren gaan verloren. Het is gewenst om te clusteren per individuele basisoorzaak omdat contextvariabelen in een bepaald type fout een andere rol kunnen hebben, een contextvariabel "overdracht dienst" kan in combinatie met een technische- een andere betekenis hebben dan in combinatie met organisatorische fout. Desondanks zouden de twee basisoorzaken dezelfde richting opgaan in die dimensie van het contextvariabel.

Toelichting CompLearn unrooted-binary-tree In deze visualisatie is de distance-matrix gebruikt die gehanteerd wordt in de K-Means clustering implementatie en zijn distances op basis van de Euclidische afstand (zie subsectie 5.1.1 paragraaf "Toepassing"). CompLearn's "clustering by compression" is niet gebruikt, alleen de maketree functionaliteit die een unrooted binary tree opbouwt (een boom waar elke knoop/parent maximaal twee childs kan hebben en waar geen specifieke (begin/root)knoop is). De output is te vinden in figuur 7.5.

Visualisatie techniek	**Resultaat**
Scatterplots	Visualiseren van clusters in een 2D of 3D weergave is gedaan door contextvariabelen te aggregeren, de indeling van basisoorzaken in clusters wordt echter niet bepaald door de aggregatie en dat kan daarom zorgen voor onlogisch gekleurde clusters. Clusters zijn op het niveau van alle basisoorzaken niet sterk genoeg, op het niveau van individuele classificaties beter. De kwaliteit- en het gebruik van de contextvariabelen zorgt nog voor door elkaar heen lopende clusters (oftewel geen clusters).
CompLearn maketree	De visualisatie neemt veel tijd in beslag, echter na een uur of twee is de output al bruikbaar, daarna worden slechts minimale verbeteringen doorgevoerd. Deze visualisatie kan wel laten zien dat bepaalde clusters *i.e.*, de blauwe en paarse, eigenlijk één groep zijn en door K-Means kunstmatig opgehakt zijn omdat de K (aantal clusters) op 7 was ingesteld. (Het maketree programma gebruikte dezelfde distance matrix als K-Means, de kleuren van K-Means zijn echter later toegevoegd.)

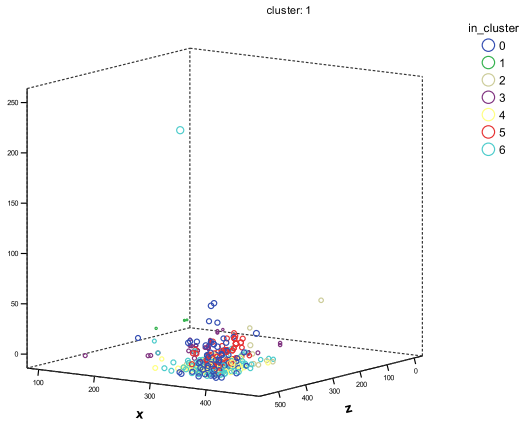
Tabel 7.3: Resultaten per K-Means cluster visualisatie samengevat.

Clustering techniek	**Resultaat**
K-Means	K-Means clustering kon niet van worden aangetoond dat dit te gebruiken is op de gegevensbron van 2006, waar het contextvariabelenmodel van 2006 wordt gehanteerd. Het is mogelijk dat met de kwalitatief hopelijk betere contextvariabelen—die opnieuw zijn afgesproken (mede op basis van voortschrijdend inzicht van het voorgaande model)—de clustering wel betere resultaten geeft. Daarnaast is het nog niet uitgeprobeerd in een situatie waar meerdere gegevensbronnen gebruikt worden.
CompLearn (compressie)	De resulterende output zou als een alternatief profiel gebruikt kunnen worden ten behoeve van het vinden van overeenkomstige instituten *etc.*, zie in sectie 5.1.3.

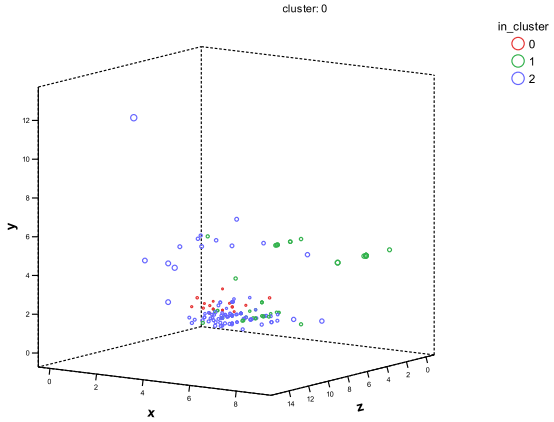
Tabel 7.4: Resultaten clustering technieken samengevat.

7.3. Text-mining analyse

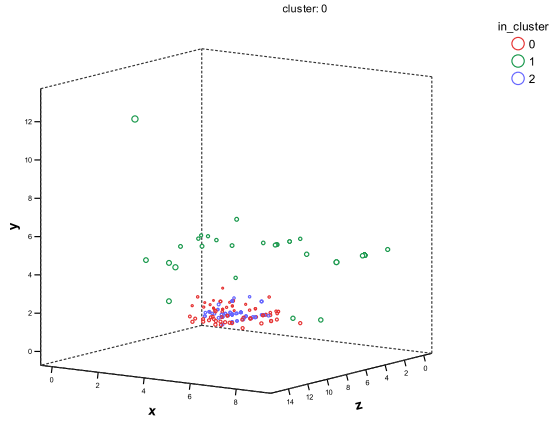
De resultaten uit sectie 5.2 zijn samengevat in tabel 7.5.



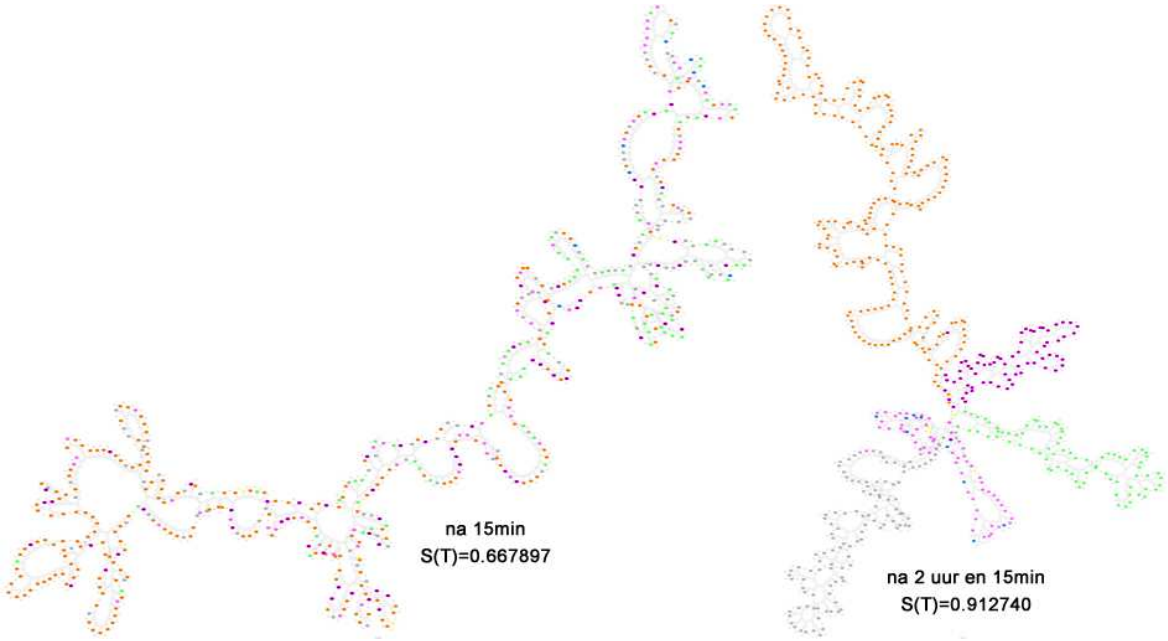
Figuur 7.1: 3D scatterplot over alle basisoorzaken drie clusters op basis van contextvariabelen.



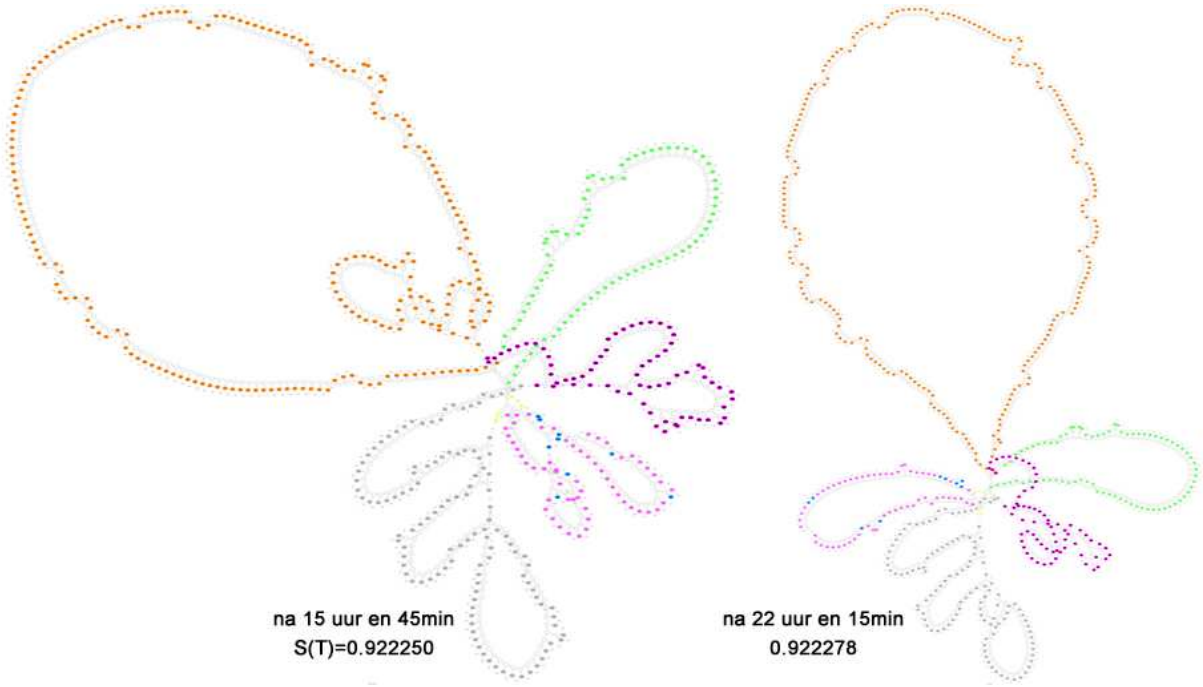
Figuur 7.2: 3D scatterplot over TD basisoorzaken drie clusters op basis van contextvariabelen.



Figuur 7.3: 3D scatterplot over TD basisoorzaken drie clusters op basis van contextvariabelen.



Figuur 7.4: Unrooted binary tree over alle basisoorzaken zeven clusters.



Figuur 7.5: Unrooted binary tree over alle basisoorzaken zeven clusters.

Text-mining techniek	Resultaat
Pathologie codering systeem	Niet toepasbaar op vrije tekst meldingen in verband met, kwaliteit van de tekst (verschillende disciplines doen meldingen met verschillend vocabulaire, *etc.*) en kwantiteit van de tekst (weinig woorden ten opzichte van de pathologierapporten).
LSI similar terms	Veelbelovende resultaten, is niet uitvoerig getest en de techniek vereist nog enige verfijning (onder andere in het stemming algoritme).
LSI clustering	Toepassing van Latent Semantic Indexing voor clustering zijn ook nog steeds dezelfde verfijningen nodig. Clustering is wel uitgevoerd, maar de resultaten zullen in de toekomst nog beoordeeld moeten worden.

Tabel 7.5: Resultaten per visualisatie clusters samengevat.

7.4. Kansberekening

De resultaten uit hoofdstuk 4 en hoofdstuk 6 (sectie 6.3) zijn samengevat in tabel 7.6. ■

Toepassing	Resultaat
Poisson verdeling	Is geschikt per week.
K-S goodness-of-fit	Is geschikt om te beoordelen of de metingen de Poisson verdeling volgen.
Chi kwadraat goodness-of-fit	Is *niet* geschikt, test is te gevoelig en geeft foute uitkomsten.
Chi kwadraat op groepen	Is geschikt, het aantal metingen moet echter niet te klein zijn.

Tabel 7.6: Resultaten kansberekening samengevat.

8 Evaluatie en Conclusie

Ter afsluiting wordt eerst teruggekoppeld naar de probleemstelling in sectie 8.1. Hoofdstukken 4, 5 en 6 lieten wel al zien hoe antwoord gegeven wordt op de subonderzoeksvragen in die hoofdstukken. In dit hoofdstuk wordt op een algemenere wijze teruggekoppeld.

Gedurende het onderzoek zijn veel potentiële onderzoeksrichtingen onderkend, die worden gegeven in sectie 8.2. Hier kan in de toekomst onderzoek naar gedaan worden.

Het hoofdstuk sluit af met een evaluatie van de onderzoeksmethode zoals deze gedefinieerd is in hoofdstuk 3. Hiermee is dit onderzoek van probleemstelling tot conclusie afgesloten, met een hoop pointers voor toekomstig onderzoek.

8.1. Terugkoppeling probleemstelling

In de radiotherapie wordt de PRISMA methode gebruikt om ten gunste van patiëntveiligheid de juiste verbetermaatregelen (zie figuur A.3, de interventiematrix in appendix A) te nemen op één of meerdere processen. Aan het nemen van verbetermaatregelen gaan beslissingen vooraf die weer gebaseerd zijn op informatie. Uit benchmarking kan extra informatie volgen en hopelijk de "decision space" vergroten en daarmee de kans op het nemen van de meest efficiënte verbetermaatregel. Op effectiviteit kunnen we ons niet richten omdat de mate van ernstigheid van de onderliggende melding en dergelijke niet bekend zijn, de PRISMA methodiek is er juist op gericht dat naast het ingrijpen op basis van individuele (eventueel de meest ernstige) incidenten ook ingrepen gedaan worden op het algemenere PRISMA profiel (en daarmee hopelijk de structurele onderliggende oorzaken aan te pakken).

Uitgaande van de hoofdonderzoeksvraag *"Hoe kunnen de PRISMA-analyse gegevens voor het Radiotherapeutisch netwerk gebruikt worden voor benchmarking op basis van data-analyse en data-mining?"*. Is mede-uitgaande van onze focus op efficiëntie het essentieel om inzichtelijk te krijgen hoe groot de kans is dat de metingen de werkelijkheid representeren. Alsmede hoeveel de werkelijkheid er met een bepaalde zekerheid eventueel vanaf kan liggen (foutmarge). Hiervoor is de theoretische verdeling (Poisson verdeling) gevonden en een goodness-of-fit test (Kolmogorov-Smirnov) om te determineren wanneer de Poisson gebruikt kan worden. Hiervan is ten behoeve van "data-analyse" aangetoond dat het werkt—zie resultaten sectie 7.4—en is subonderzoeksvraag *Sub-1* beantwoord.

Naast deze wens om juist geïnformeerd te worden is het ook gewenst om volledig geïnformeerd te worden. Interventies nemen op basis van typen oorzaken is één kant van het verhaal, combinaties van typen oorzaken kunnen ook voorkomen, na combinaties wellicht nog complexere structuren en hier biedt data-mining uitkomst. Hiervan is aangetoond dat clustering in de vorm van K-Means op de huidige data niet werkt, zie resultaten in sectie 7.2. Classificaties en contextvariabelen zijn ook maar één kant van het verhaal als nagegaan wordt dat de tekstuele meldingen (vrije tekst, meerkeuzevragen, ...) en PRISMA analysen (vrije tekst in alle oorzaken en niet alleen basisoorzaken) allerlei gedetailleerde informatie bevatten. De analisten gebruikten tijdens de evaluatie van het prototype ook vaak de originele basisoorzaakomschrijvingen als laatste validatiemiddel. Hier biedt text-analyse uitkomst en daarin is aangetoond dat Latent Semantic Indexing werkt, zie resultaten sectie 7.3. Aanvullend toekomstig onderzoek is hierin wel nodig

(zie sectie 8.2). Hiermee is antwoord gegeven op *Sub-2*.

Het prototype implementeert in eerste instantie de huidige werkwijze, zie resultaten sectie 7.1. Met deze uitkomsten is antwoord gegeven op *Sub-3.1*. De mate waarin de resultaten tot dusver bijdragen aan ten eerste het deductieproces (dat wil zeggen de analyse zoals de analist deze uitvoert zonder benchmarking met andere partijen) is concreet de mogelijkheid tot: significantieberekeningen voor metingen over de tijd; significantieberekeningen voor verschil tussen groepen; clustering over de landelijke database (niet clustering per instituut en dan vergelijken); de toepassing van LSI in die zin dat het in tabelvorm verbanden op laag niveau (van concepten) kan presenteren (de concepten zullen de analist meer zeggen dan verbanden in contextvariabelen omdat deze een stuk concreter zijn), ten tweede het benchmarkingproces (waar wel vergeleken wordt met andere partijen) is concreet de mogelijkheid tot: combinaties van typen oorzaken of andere complexere selecties. Hiermee is antwoord gegeven op *Sub-3.2* met als kanttekening dat de toegevoegde waarde nog niet in de mate is aangetoond waarin dit zou moeten kunnen, hier is toekomstig onderzoek nog voor vereist.

8.2. Toekomstig onderzoek

Twee grondredenen voor het *niet* gebruiken van de vrijetekst zijn kunnen worden weerlegd. De eerste is dat de PRISMA classificaties en contextvariabelen meer zijn gestandaardiseerd. Wat echter blijkt is dat de tekst juist ideale uitkomst biedt om bepaalde vermoedens wat betreft nog specifiekere oorzaken te verifiëren. De tweede grondreden is dat de teksten te veel informatie blootleggen ten opzichte van andere partijen. Ook dit is niet een probleem want de clustering op basis van LSI biedt de mogelijkheid toch te putten uit de informatie, en deze zelfs visueel weer te geven, *zonder* dat er enige oorzaakgevolg informatie wordt vrij gegeven. De woorden worden namelijk samen gepresenteerd in een tabelvorm, en de enige relatie die blootgelegd wordt hiertussen is dat ze volgens bepaalde algoritmes met elkaar te maken hebben. Dit is de rede dat volgens mijn inschatting de text-analyse richting de meest waardevolle resultaten zal opleveren. Clustering op basis van concepten heeft immers ook het voordeel niet afhankelijk te zijn van een (goed functionerend) classificatie/contextvariabelen-systeem.

Een opsomming van een aantal concrete richtingen die in dit onderzoek onderkend zijn:

- Wat is de toegevoegde waarde[1] van LSI na het gereedmaken voor Nederlandse teksten?
- Hoe kan LSI assisteren in het opbouwen van een fijnere rubricering in de vorm van trefwoorden?
- Hoe kan de K-S goodness-of-fit test gebruikt worden om meldgedrag te compenseren?
- Hoe verhoudt een niet-statisch contextvariabelenmodel *i.e.*, gebruik van *tags*, zich tot het huidige statische model?
- Is de 3D scatterplot visualisatie te verbeteren door niet te aggregeren op contextvariabelen maar door op een andere manier de richtingen te bepalen van de vectoren, *e.g.*, alle richtingen in de 3D ruimte verdelen over alle contextvariabelen?

Een opsomming van een aantal resultaten die zullen voortvloeien uit het vervolg van dit onderzoek:

- Toegevoegde waarde[2] van Clustering op een landelijke database met meerdere instituten [3].

[1] De toegevoegde waarde is gedefinieerd als door het systeem "gevonden uitkomsten" ("opmerkelijke observaties") die nog *niet* bekend waren en die *niet* door de analist ontdekt hadden kunnen worden zonder het systeem (vanwege tijd en moeite benodigd zonder systeem).

[2] *ibid.*

[3] De pilot-in gebruik neming- van het GreCom VMS en de software die gedurende dit onderzoek ontwikkeld is zal de nieuwe set van Contextvariabelen gaan gebruiken, 1 juli 2008 zal hier officieel mee gestart worden met drie instituten. Vanaf 1 januari 2009 zullen gefaseerd de overige RT instituten toegevoegd worden aan het netwerk. Gedurende de pilot zal daarom blijken of K-Means clustering op basis van betere contextvariabelen meer oplevert dan tot nu toe getoetst in het hoofdexperiment.

- Toegevoegde waarde van Clustering op een landelijke database met verbeterde contextvariabelen.
- Toegevoegde waarde van Control Charts op een landelijke database waar (waarschijnlijk) geen overschaduwende en *expliciete* gebeurtenissen (zoals de verhuizing) plaatsvinden[4].

8.3. Evaluatie onderzoeksmethode

Omdat dit onderzoek een exploratief onderzoek is geweest zijn veel richtingen tegelijk aangesneden. Daardoor heb ik tegen het einde van het onderzoek niet nog extra veel diepgang kunnen aanbrengen in één van de onderwerpen. Een concreet voorbeeld met betrekking tot extra diepgang (deels op basis van voortschrijdend inzicht): de K-Means clustering toepassing combineren met de kansberekening in een systeem dat dan zelf moet kunnen bepalen wat interessante vindingen zouden moeten zijn in de PRISMA gegevensbron. Omdat te veel onderwerpen zijn aangesneden waren, en de verwerking hiervan in één verhaal veel tijd kostte, was hier geen ruimte meer voor.

Achteraf was het genoemde voorbeeld te doen geweest, ondanks dat gebleken is dat de gegevensbron op dit moment niet geschikt is. Niet geschikt in die zin dat K-Means clustering geen aantoonbaar bruikbare resultaten opleverde, precies de rede waarom in dit onderzoek ervoor gekozen is dit nog niet te doen. Hiervoor kan als oplossing een gecontroleerde kunstmatige gegevensbron gegenereerd worden. Een gegevensbron waarin willekeurig allerlei fouten worden gegenereerd, waarvan ook een paar specifieke op voorhand gedefinieerde fouten wat minder willekeurig en vaker voor laten komen. Dan kunnen de geïmplementeerde formules en algoritmen hierop uitgeprobeerd worden en dan zullen deze "fouten" gevonden moeten worden. Factoren als stimuleren van meldgedrag kunnen ook gesimuleerd worden, *etc.* Daarna kan alsnog overgegaan worden tot het uitproberen van dezelfde techniek op de gegevensbron. Op deze manier had meer diepgang in de onderwerpen gerealiseerd kunnen worden.

Tot slot, terugkomende op het gebruik van benchmarking in de Radiotherapie in het algemeen. Het is heel interessant om bij een extern instituut te kijken dat een andere procesinrichting heeft en te constateren dat daar bepaalde problemen niet veel voorkomen. Een statistisch stabiel profiel wordt bepaald door de karakteristieken van de afdeling, werkeenheid of instituut. Tenzij twee processen echt identiek zijn (op papier *e.g.*, een gedefinieerd proces) worden er meer appels met peren vergeleken en geen appels met appels. Het hoeft dus niet uit te maken of er appels met peren vergeleken worden.

Bij de MAASTRO zijn ernstige incidenten zeer schaars, desondanks hoop ik dat het systeem in de toekomst zal bij dragen aan het nog verder terugbrengen van incidenten in de radiotherapie. Door de technieken die Software Engineering, kan implementeren het systeem de mogelijkheden kan benutten die normaal gesproken nooit benut hadden kunnen worden. Ik ben ook van mening dat de informatie die volgt uit vergelijking en samenvoeging tussen de RT instituten de grootste nog niet geputte (potentiële) bron van informatie is. ■

[4] Met niet-expliciet bedoel ik dat ingrijpende gebeurtenissen altijd kunnen voorkomen, alleen deze zijn dan niet al op voorhand te voorzien, *e.g.*, problemen met luchtcirculatie of de electriciteitsvoorziening in een bepaalde vleugel van het gebouw.

Bibliografie

[1] Baneyx, A., Charlet, J., and Jaulent, M.-C. Building medical ontologies based on terminology extraction from texts: an experimentation in pneumology. Stud Health Technol Inform. 2005;116:659-64. PMID: 16160333.

[2] Birnbaum, A. Some procedures for comparing poisson processes or populations. *Biometrika 40*, 3/4 (1953), 447–449.

[3] Cilibrasi, R., and Vitanyi, P. M. B. A new quartet tree heuristic for hierarchical clustering, 2006.

[4] de Bruijn, L., Hasman, A., and Arends, J. Supporting the classification of pathology reports: comparing two information retrieval methods. Comput Methods Programs Biomed. 62(2):109-13. 2000. PMID: 10764937.

[5] Emslie, S., et al. *Improving Patient Safety: Insights from American, Australian and British healthcare*. ECRI and contributors, 2002.

[6] George, M. L. *Lean Six Sigma for Service: How to Use Lean Speed and Six Sigma Quality to Improve Services and Transactions*. McGraw-Hill Professional, 2003.

[7] Hasman, A., de Bruijn, L., and Arends, J. Evaluation of a method that supports pathology report coding. Methods Inf med 2001;40(4):307-14. PMID: 11552341.

[8] HR., B. An introduction to benchmarking in healthcare. PMID: 10139084 [PubMed - indexed for MEDLINE].

[9] Jeong, S., and Kim, H.-G. Ontology based adverse event reporting system architecture. Center for Healthcare Ontology R&D, Seoul National University (Korea), 2006.

[10] Kloos, M. Comm.unities.of.prac.tice 2.0 - how blogs, wikis, and social bookmarking offer facilities that support learning in practice in communities of practice. Master's thesis, University of Amsterdam, 2006.

[11] McDonald, K. M., et al. Measures of patient safety based on hospital administrative data. the patient safety indicators. AHRQ Publication No. 02-0038. August 2002.

[12] Mcellin, E. B. Stochastic modeling in health insurance. In *RECORD, Volume 31, No.2, New Orleans Health/Pension Spring Meeting, June 15-17, 2005. Session 76PD* (2005). url `http://www.soa.org/library/proceedings/record-of-the-society-of-actuaries/2000-09/2005/june/rsa05v31n276pd.pdf`.

[13] Medvidovic, N. On the role of middleware in architecture-based software development. In *SEKE '02: Proceedings of the 14th international conference on Software engineering and knowledge engineering* (New York, NY, USA, 2002), ACM Press, pp. 299–306.

[14] Medvidovic, N., Mehta, N. R., and Mikic-Rakic, M. A family of software architecture implementation frameworks. In *WICSA 3: Proceedings of the IFIP 17th World Computer Congress - TC2 Stream / 3rd IEEE/IFIP Conference on Software Architecture* (Deventer, The Netherlands, The Netherlands, 2002), Kluwer, B.V., pp. 221–235.

[15] Mols, B. Nrc artikel: Meesterlijk complex. Interview met Paul Vitányi, online beschikbaar: `http://homepages.cwi.nl/~paulv/papers/nrc07.pdf`. dd. 8 sept. 2007.

[16] Ng, H. K. T., Gu, K., and Tang, M. L. A comparative study of tests for the difference of two poisson means. *Comput. Stat. Data Anal. 51*, 6 (2007), 3085–3099.

[17] NHS. Nhs performance indicators.

[18] NIVEL. Onbedoelde schade in nederlandse ziekenhuizen (publieksamenvatting). `http://www.nivel.nl/pdf/onbedoelde-schade-in-nederlandse-ziekenhuizen-publieksamenvatting-2007.pdf`.

[19] Persse, J. R. *Process Improvement Essentials: CMMI, Six SIGMA, and ISO 9001.* O'Reilly Media, Inc., 2006.

[20] Pettitt, A. N., et al. The kolmogorov-smirnov goodness-of-fit statistic with discrete and grouped data. Technometrics, Vol. 19, No. 2 (May, 1977), pp. 205-210.

[21] Pidcock, W. What are the differences between a vocabulary, a taxonomy, a thesaurus, an ontology, and a meta-model? `http://www.metamodel.com/article.php?story=20030115 211223271` 2003, 2003.

[22] Pyzdek, T. When in doubt, get the x chart out. article, available at `http://www.qualitydigest.com/feb98/html/spctool.html`.

[23] Pyzdek, T. *The Six Sigma Handbook, Revised and Expanded: A Complete Guide for Greenbelts, Blackbelts and Managers at All Levels.* McGraw-Hill Professional (2nd ed.), 2003.

[24] RC, C., et al. Benchmarking applied to health care. PMID: 8044218 [PubMed - indexed for MEDLINE].

[25] Reijnders-Thijssen, P. Gezamenlijk patiëntveiligheidstraject in de radiotherapie. 2005.

[26] Rozovsky, F. A., et al. *The Handbook of Patient Safety Compliance: A Practical Guide for Health Care Organizations.* Jossey-Bass (March 21, 2005).

[27] S, L., et al. Benchmarking: finding ways to improve. PMID: 8044220 [PubMed - indexed for MEDLINE].

[28] Tan, J., et al. *E-Health Care Information Systems: An Introduction for Students and Professionals.* Jossey-Bass (April 25, 2005).

[29] Tan, P.-N., Steinbach, M., and Kumar, V. *Introduction to Data Mining, (First Edition).* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.

[30] Taylor, R. N., Medvidovic, N., Anderson, K. M., E. James Whitehead, J., and Robbins, J. E. A component- and message-based architectural style for gui software. In *ICSE '95: Proceedings of the 17th international conference on Software engineering* (New York, NY, USA, 1995), ACM Press, pp. 295–304.

[31] U.S. Navy Pacific Fleet. Basic tools for process improvement - module 10: Control chart. public domain, available at `http://www.balancedscorecard.org/resources/wpapers.html`.

[32] U.S. Navy Pacific Fleet. Handbook for basic process improvement. public domain, available at http://www.balancedscorecard.org/resources/wpapers.html.

[33] van der Schaaf, T., and Habraken, M. Prisma-medical, a brief description.

[34] van Everdingen, J., Smorenburg, S., et al. *Praktijkboek patiëntveiligheid*. Bohn Stafleu van Oghum (Houten, 2006).

[35] Vermaat, M. B. *Statistical process control in non-standard situations*. PhD thesis, 2006. pdf document http://dare.uva.nl/document/36871.

[36] voor de Gezondheidszorg, I. *Het resultaat telt: Prestatie-indicatoren als onafhankelijke graadmeter voor de kwaliteit van in ziekenhuizen verleende zorg*. Rapport, IGZ 07-48; oplage 1600.

[37] Weber, A. Poisson processen. Vrije Universiteit Amsterdam, Opleiding Wiskunde - Vak Poisson Processen. Januari 2003.

[38] Website. Engineering Statistics Handbook - 7.2.1.2. Kolmogorov- Smirnov test http://www.itl.nist.gov/div898/handbook/prc/section2/prc212.htm dd. 6.feb.2008.

[39] Website. The Semantic Indexing Project Documentation, Chapter 3: Pruning/Refining Searches. http://www.knowledgesearch.org/doc/examples.html.

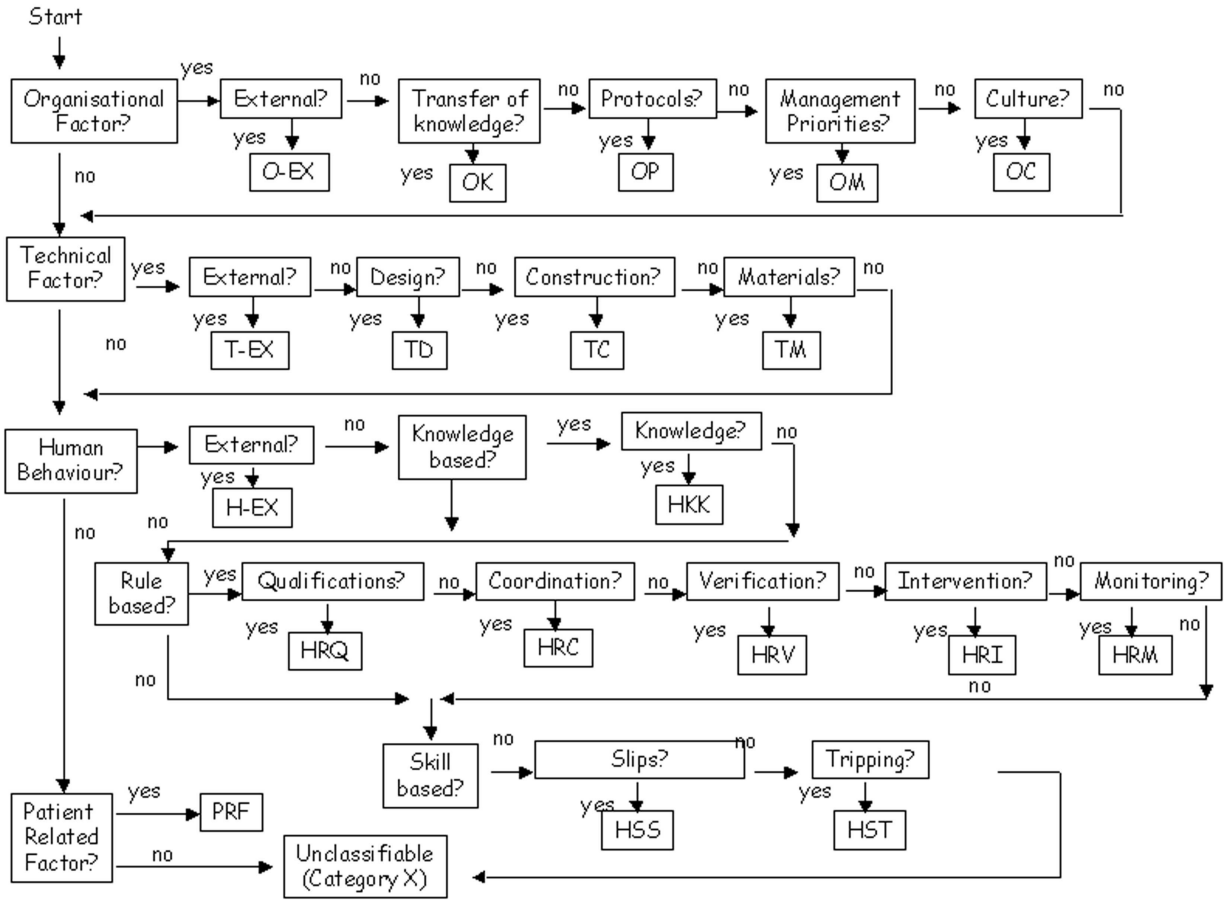
[40] Website. Engineering Statistics Handbook - 1.3.6.6.6. Chi-Square Distribution http://www.itl.nist.gov/div898/handbook/eda/section3/eda3666.htm dd. 6.feb.2008.

[41] Willems, R. Hier werk je veilig, of je werkt hier niet. In *Sneller Beter - De veiligheid in de zorg* (Carel van Bylandtlaan 30, 2596 HR Den Haag, 2004).

[42] Yu, C., et al. Patterns in unstructured data: Discovery, aggregation, and visualization. National Institute for Technology and Liberal Education (NITLE). url: http://www.seobook.com/lsi/cover_page.htm 2002.

A PRISMA methode bijlagen

De beslisboom in te vinden in figuur A.1 [1] het volledige model in figuur A.2 [2] en de bijbehorende interventiematrix in figuur A.3 [3].



Figuur A.1: PRISMA Beslisboom Eindhoven Classificatie Model.

[1] bron: `http://www.dcs.gla.ac.uk/~johnson/papers/case_based_reasoning/`.
[2] bron: `http://www.who-icps.org/resources/PRISMA-Medical.pdf`.
[3] *ibid.*

Table 1. Categories of the Eindhoven Classification Model: medical version [MERS TM, 2001; van Vuuren et al., 1997].

		Code	Category	Definition
Technical		T-EX	External	Technical failures beyond the control and responsibility of the investigating organisation.
		TD	Design	Failures due to poor design of equipment, software, labels or forms.
		TC	Construction	Correct design, which was not constructed properly or was set up in inaccessible areas.
		TM	Materials	Material defects not classified under TD or TC.
Organisational		O-EX	External	Failures at an organisational level beyond the control and responsibility of the investigating organisation, such as in another department or area (address by collaborative systems).
		OK	Transfer of knowledge	Failures resulting from inadequate measures taken to ensure that situational or domain-specific knowledge or information is transferred to all new or inexperienced staff.
		OP	Protocols	Failures relating to the quality and availability of the protocols within the department (too complicated, inaccurate, unrealistic, absent, or poorly presented).
		OM	Management priorities	Internal management decisions in which safety is relegated to an inferior position when faced with conflicting demands or objectives. This is a conflict between production needs and safety. An example of this category is decisions that are made about staffing levels.
		OC	Culture	Failures resulting from collective approach and its attendant modes of behaviour to risks in the investigating organisation.
Human		H-EX	External	Human failures originating beyond the control and responsibility of the investigating organisation. This could apply to individuals in another department.
	Knowledge-based behaviour	HKK	Knowledge-based behaviour	The inability of an individual to apply their existing knowledge to a novel situation. Example: a trained blood bank technologist who is unable to solve a complex antibody identification problem.
	Rule-based behaviour	HRQ	Qualifications	The incorrect fit between an individuals training or education and a particular task. Example: expecting a technician to solve the same type of difficult problems as a technologist.
		HRC	Coordination	A lack of task coordination within a health cares team in an organisation. Example: an essential task not being performed because everyone thought that someone else had completed the task.
		HRV	Verification	The correct and complete assessment of a situation including related conditions of the patient and materials to be used *before* starting the intervention. Example: failure to correctly identify a patient by checking the wristband.
		HRI	Intervention	Failures that result from faulty task planning and execution. Example: washing red cells by the same protocol as platelets.
		HRM	Monitoring	Monitoring a process or patient status. Example: a trained technologist operating an automated instrument and not realizing that a pipette that dispenses reagents is clogged.
	Skill-based behaviour	HSS	Slips	Failures in performance of highly developed skills. Example: a technologist adding drops of reagents to a row of test tubes and than missing the tube or a computer entry error.
		HST	Tripping	Failures in whole body movements. These errors are often referred to as "slipping, tripping, or falling". Examples: a blood bag slipping out of one's hands and breaking or tripping over a loose tile on the floor.
Other factors		PRF	Patient related factor	Failures related to patient characteristics or conditions, which are beyond the control of staff and influence treatment.
		X	Unclassifiable	Failures that cannot be classified in any other category.

Table 2. Classification of recovery factors [Personal communication with van der Schaaf, March 2005].

	Planned	Not planned
Human	P-H	NP-H
Technical	P-T	NP-T
Organisational	P-O	NP-O
Patient-related	(P-PRF)	NP-PRF
Unclassifiable		NP-X

Figuur A.2: PRISMA Eindhoven Classificatie Model.

Table 3. Classification/Action Matrix [Personal communication with van der Schaaf; March 2005].

Classification code	Technology / Equipment	Procedures	Information and Communication	Training	Motivation	Escalation	Reflection
T-EX						×	
TD	×						
TC	×						
TM	×						
O-EX						×	
OK						×	
OP		×					
OM						×	
OC							×
H-EX						×	
HKK			×		**NO**		
HRQ				×			
HRC				×			
HRV				×			
HRI				×			
HRM				×			
HSS	×				**NO**		
HST	×				**NO**		
PRF[1]							
X							

[1]If particular patient related factors (such as language problems) that cannot be prevented by the patients themselves recur, then these problems should be solved at an organisational level (i.e. escalation).

The following classes of actions are distinguished:

- Technology/Equipment: redesigning of hardware, software or interface parts of the manmachine system.
- Procedures: completing or improving formal and informal procedures.
- Information and communication: completing or improving available sources of information and communication structures.
- Training: improving (re)training programmes for skills needed.
- Motivation: increasing the level of voluntary obedience to generally accepted rules by applying principles of positive behaviour modification.
- Escalation: handling the problems at a higher organisational level.
- Reflection: evaluating the current way of behaving regarding safety.

In the column “motivation” “NO” has been placed three times because it is a common error of management to motivate (or punish) employees to prevent knowledge-based errors and skill-based errors from happening.

The Classification/Action Matrix should not (always) be followed literally. Which measures are necessary is of course completely dependent on the organisation and the nature of the incidents. Therefore it is important to register context factors too. These context factors answer questions as: who?, what?, where?, and when?.

Figuur A.3: PRISMA Interventie Matrix.

B Prototype architectuur omschrijving

B.1. Architecture constraints

Afhankelijkheid voornamelijk met een uitgebreid framework (standaardfunctionaliteit) dat bestaat uit PHP 4 code (versie 5 compatible, 839.433 regels code per 24 december 2007 exclusief comments (LOC) gemeten met StatCVS v0.2.2. [1]. Dit is exclusief modules zoals de benchmark, meldingen, *etc.* [2]). Gebruikers en groepen + rechten management, (geïntegreerd met) database layer om de database heen, (geïntegreerd met) GUI componenten [3]. Al met al om sneller applicaties te kunnen ontwikkelen. *Constraints* opgelegd door de klant en architectuur framework: LAMP architectuur (*i.e.*, Linux+Apache+{MySQL|MSSql|Oracle}+ {PHP≥4} in ons geval.), dus web applicaties, geen plugins (zoals Flash, ActiveX, Java applets, et.al.) en hoofdzakelijk Internet Explorer ≥ 6.0. *Karakteristieken* die onontkoombaar zijn: Stateless (d.w.z. uniedirectionele communicatie in principe vanuit de client), late binding (d.w.z. fouten doen zich vaak pas "at runtime" voor) reflectieve programmeertaal (refactoring daardoor ook moeilijker dan wanneer men striktere typechecking heeft [4]).

B.2. Architecture (quality) requirements

Scalability en modifiability zijn de twee meest belangrijke quality attributes. Hiernaast worden hoge eisen gesteld aan de performance van systeem.

Modifiability Vanuit een prototype is een enorme verscheidenheid aan mogelijkheden (algoritmen, visualisaties) die geïmplementeerd kunnen worden. De zekerheid *of* de implementatie zal plaatsvinden, laat staan de volgorde van implementatie, staat niet vast. Het is daarom belangrijk dat gedurende de evolutie van de software de gewenste uitbreidingen gemaakt kunnen worden.

Scalability We hebben te maken met een prototype dat begint met twee benchmarkpartijen. De potentie van het systeem is een uitbreiding sowieso naar het hele radiotherapeutische netwerk (> 10 instellingen). Rekening wordt gehouden met nog generieker uitbreidingen waardoor wellicht ook *niet*-radiotherapeutische instituten met elkaar kunnen gaan benchmarken. Al naar gelang het aantal instituten waarmee vergeleken wordt groeit stijgt voor een aantal algoritmen (die *i.e.*, euclidische afstanden berekenen tussen alle meldingen) de complexiteit (waarschijnlijk) exponentieel. Het loskoppelen van componenten die veel CPU opeisen wordt daarom misschien steeds gewenster. [13]

[1] Url: `http://statcvs.sourceforge.net/`.

[2] Het aantal regels code waaruit het benchmarkpunt prototype bestaat per 11 december 2007 uit 30.999 LOC. De meldingen module bestaat uit 150.274 LOC per 24 december 2007.

[3] Interaction Design is verantwoordelijk voor een logisch gebruikersconcept.

[4] *i.e.*, een "Extract Method" refactoring vereist al veel meer oplettendheid in welke variabelen van belang zijn. De IDE kan geen volledige statische analyse doen om hierin te ondersteunen op een manier zoals Eclipse dat bijvoorbeeld kan voor JAVA (url: `http://www.eclipse.org/`).

Performance Als ideale maat wordt een typische C of C++ desktop applicatie gehanteerd. Deze maat is idealistisch en niet haalbaar voor een webapplicatie vanwege beperkt geheugen via browser, e.g. code wordt geïnterpreteerd en nooit volledig gecompileerd. Performance is belangrijk, als het tempo van de applicatie laag ligt kan er in een gegeven tijdsbestek minder onderzocht worden met het systeem.

B.2.1. Problemen in het realiseren quality requirements

Ontstaan door *constraints* (i): grafische weergaven zonder plugins zoals Flash (zoals gebruikt in Google Analytics [5]) zijn aanzienlijk trager. Alternatieven zijn: (1) genereren van plaatjes + sturen van server naar client en dit is aanzienlijke overhead in vergelijking met flash waar alles client side gerenderd wordt; (2) genereren van plaatjes m.b.v. canvas (native ondersteund in Firefox, kunstmatig in IE beschikbaar gemaakt door Google [6]) dit is sowieso allemaal erg traag omdat het met javascript gerenderd wordt. Ontstaan door *karakteristieken* (ii): met elk request worden alle objecten opnieuw opgebouwd in het geheugen op de server, dit omvat veel parse-, initialisatie- en executie- tijd. Ontstaan door *afhankelijkheden* (iii): GUI componenten die op een generieke manier de interface genereren zijn flexibel opgebouwd, met als gevolg dat in een bepaalde interface meerdere requests gedaan moeten worden. Concreet kon dit kon oplopen tot ongeveer (afhankelijk van de view) 4 à 6 seconden gemiddelde laadtijd. Een specifieke view nam gemiddeld 6343ms client+server laadtijd in beslag, waarvan 4295ms servertijd is. Deze view bestond uit 8 requests, dus per pagina een halve seconde, waarvan ongeveer 300ms parse+initialisatie tijd omvatte. De gemiddelden zijn berekend op basis van 5 minuten lang de volledige view constant opnieuw opbouwen in Internet Explorer (47x).

B.3. Synthese d.m.v. Architectuur

Oplossingen problemen De problemen m.b.t. het snel kunnen renderen van visualisaties (i) zijn *niet* opgelost. Wel zijn de visualisatie componenten dusdanig opgezet dat ze in de toekomst vervangen kunnen worden door componenten die *wel* gebruik maken van plugins zoals Flash, mochten de constraints aangepast worden. De problemen m.b.t. parse-, initialisatie- en executie- tijd (ii), alsmede de problemen met de trage GUI-opbouw (iii), zijn opgelost m.b.v. een innovatieve oplossing om zoveel mogelijk objecten persistent te houden op de server. Deze oplossing is TDP genoemd en de implementatie wordt besproken in paragraaf B.3.1. De TDP oplossing dat als eerste geïmplementeerd is in het Framework, heeft de mogelijkheden wat betreft architectuur aanzienlijk vergroot.

Kwaliteit in de architectuur Gedurende de Software Construction course aan de UvA heb ik mij onder andere verdiept in Component Based Development, Service Oriented Architectures en de C2 Architectural style. Deze onderwerpen zijn destijds onderzocht binnen het kader van een kwalitatieve case; ditzelfde Framework. Dezelfde karakteristieken in acht nemende zocht ik naar een constructie dat binnen dit kader de genoemde kwaliteitsaspecten kon bewerkstelligen. De C2 Architectural style besproken in paragraaf B.3.2 kwam in aanmerking maar moest er dan wel een deel van het systeem persistent gemaakt worden. Deze voorwaarde is aanleiding geweest voor de TDP oplossing (B.3.1).

B.3.1. TDP in een notendop

De software is stateless en bij elk request moet de PHP code ge- parsed, -initialiseert en uitgevoerd worden (misschien hier en daar met wat optimalisatie door de Zend engine optimizer [7]). In figuur

[5] Url: `http://www.google.com/analytics/features.html`.
[6] Url: `http://excanvas.sourceforge.net/`.
[7] Url: `http://www.zend.com/en/products/guard/optimizer/`.

B.1 is een typisch request conceptueel visueel weergegeven. De architectuur genereert een response en bestaat abstract gezien uit JSON of HTML. De duur van het request in het blauw aangegeven is 200 milliseconden. Server load in tijd is conceptueel van boven naar beneden weergegeven (zie groene pijl).

In figuur B.2 volgt een uitgebreider plaatje namelijk hetzelfde request en respons, maar waar TDP tussen zit om de boel te versnellen. Het hele concept zal duidelijk worden na de beschrijving van twee opeenvolgende requests binnen dezelfde sessie. Het eerste request zal TDP initialiseren, het tweede en elk volgend request zal vervolgens de in request nummer één geinitialiseerte TDP server gebruiken om het request af te handelen en zelf niets meer includen/parsen/executeren.

B.3.2. C2 in een notendop

C2 is een component-based architectural style met striktere message passing. In ontwikkeling op het Computer Science Department aan de University of Southern California. Zij omschrijven de style het als volgt. Een netwerk van componenten aan elkaar gekoppeld door middel van connectoren. Communicatie tussen de componenten verloopt door het versturen van messages door de connectoren. Elk component kan zijn eigen (één of meer) controle threads hebben. Voor de volgende uitleg is het handig om figuur B.3 als ondersteuning te gebruiken.

Componenten en connectoren zijn verheven tot first-class entiteiten en hebben beide een top en bottom interface. Ze worden aan de hand van de volgende set van regels aan elkaar gekoppeld:

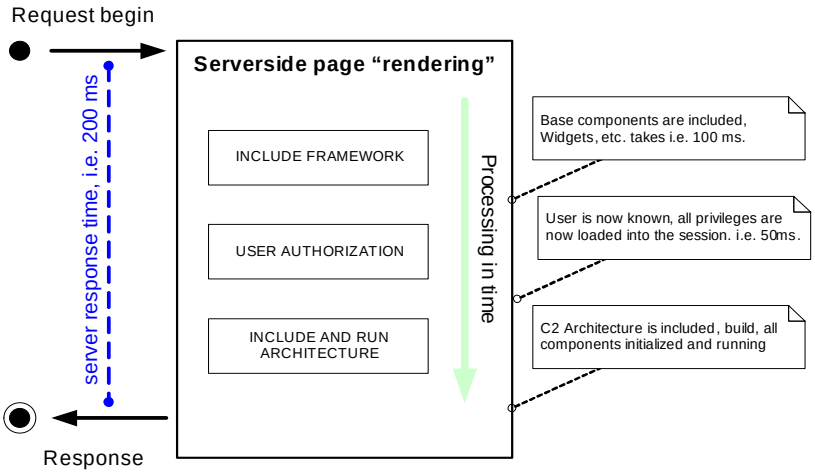
- De top van een component mag verbonden worden met de bottom van één connector;
- De bottom van een component mag alleen verbonden worden met de top van één connector;
- Er is geen limiet aan het aantal componenten of connectoren dat verbonden kan zijn aan één connector;
- Wanneer twee connectoren verbonden zijn aan elkaar moeten deze dat zijn middels de bottom van de ene met de top van de andere;

Alle communicatie verloopt door middel van message passing door de connectoren. De top-interface van een component specificeert op welke set notificaties het component reageert, en de set van requests die het verstuurt naar componenten “boven” zich in de architectuur. De bottom-interface van een component specificeert welke notificaties het component naar de componenten “onder” zich in de architectuur verstuurd, en de set van requests waar het op reageert.

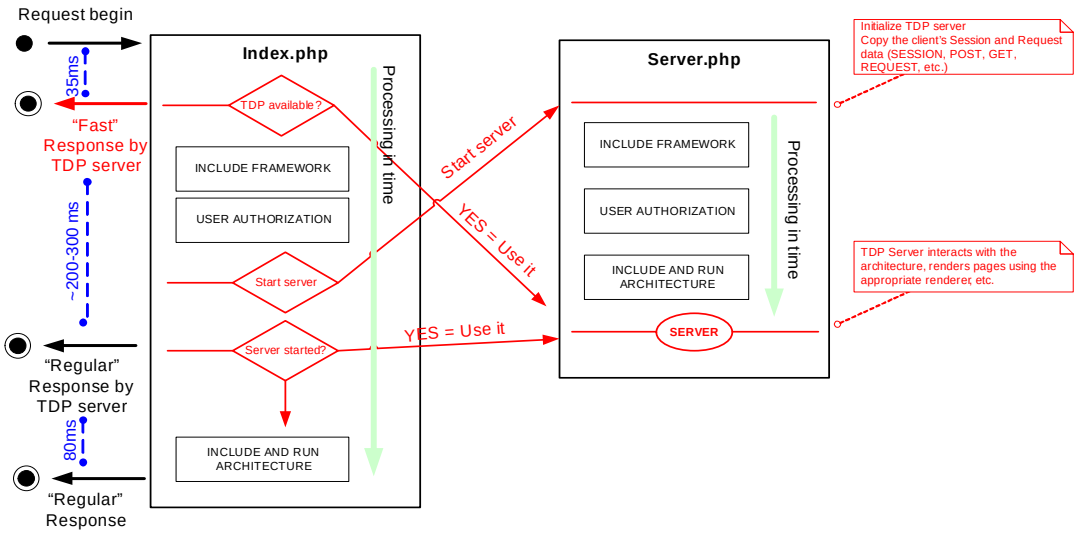
De essentie van deze stijl is *limited visibility* en *substrate independence*. Een component in de hiërarchie is zich namelijk alleen “bewust” van componenten *boven* zich en zich volledig onbewust van componenten *onder* zich. Omdat componenten zo weinig van hun omgeving weten wordt *loose coupling* bereikt. Elk component uit de hiërarchie is makkelijk te vervangen door een andere implementatie. Ook is het makkelijk om complete lagen van connectoren toe te voegen zonder dat de componenten zelf aangeraakt hoeven worden.

Middleware Nenad Medvidovic laat uitvoerig zien hoe de first-class connectoren binnen C2 verticaal en horizontaal zijn te splitsen en op een manier dat dit *geen* invloed heeft op de karakteristieken van de stijl [13]. Hij laat zien hoe gemakkelijk middleware gebruikt kan worden om het systeem te distribueren. Onafhankelijk van specifieke middleware oplossingen.

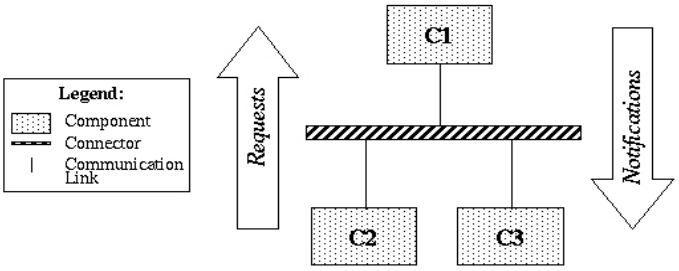
C2 in Webapplicaties “The C2 architectural style is designed to support the particular needs of applications that have a graphical user interface aspect, but it has the potential for supporting other types



Figuur B.1: Conceptuele weergave request van begin tot response terug naar client.



Figuur B.2: Conceptuele weergave request en de rol van TDP.



Figuur B.3: Conceptuele weergave C2 style.

of applications as well.", volgens het C2 team [8] . Grafische userinterfaces zijn in ons geval ook een essentieel aspect van de applicatie.

C2 en Performance Performance van de C2 stijl in combinatie met het stateless aspect is een probleem. In een voorbeeld van Medvidovic kan men zien hoe de C2 stijl toegepast kan worden [14]. Alle entiteiten van de architectuur extenden het C2 Framework. Bij het starten van de applicatie worden de componenten en connectoren aan de architectuur toegevoegd. Daarna worden de componenten aan de connectoren verbonden en eventueel connectoren ook aan andere connectoren. Vervolgens wordt de architectuur gestart en alle componenten geinitaliseerd. Als dit met elk request plaats zou moeten vinden zorgt dit voor aardig wat overhead.

B.3.3. Resultaten TDP

Het in paragraaf B.2.1 omschreven probleem dat ontstaan is door afhankelijkheden, een responstijd van 4295ms, is door standaard performance verbeteringen (profilen, optimaliseren, toetsen, evalueren, ..) tot 2751ms teruggebracht. Dit is een factor van 1.56. Vervolgens is de responstijd ervan uitgaande dat TDP constant actief is 834ms. Nog eens een verbeterfactor van 3.30 op de vorige aanpassingen [9].

B.3.4. Resultaten C2

Het geïmplementeerde C2 Framework is gebaseerd op sourcecode afkomstig uit de `src` directory van ArchStudio 3.0 (afkomstig van Universiteit van Californië). Deze bevat de Java code voor het C2 Framework alsmede alle demo applicaties. De vervolgens gemaakte PHP 5 implementatie is niet een één op één port maar een nieuw (en zeer minimalistisch) ontwerp op basis van de in de code gevonden features. D.w.z. bepaalde functionaliteiten van de stijl die ik niet direct nodig had zijn overgeslagen maar wel implementeerbaar voor in de toekomst (*i.e.*, uitgebreide message queueing). In [14] is de originele "end-user API" te vinden, en wordt verder ingegaan op de objectives van de C2 stijl en andere eigenschappen van een aantal implementaties.

[8] Url: `http://www.isr.uci.edu/architecture/c2StyleRules.html`.

[9] De metingen zijn gebaseerd op een script dat constant één view als een nieuwe sessie opent. Dus bij elk eerste request moest TDP opgestart worden, dit eerste request is altijd langzamer. De 834ms is de originele meting van 1212ms minus het verschil van steeds het eerste request t.o.v. het gemiddelde van de overige requests.

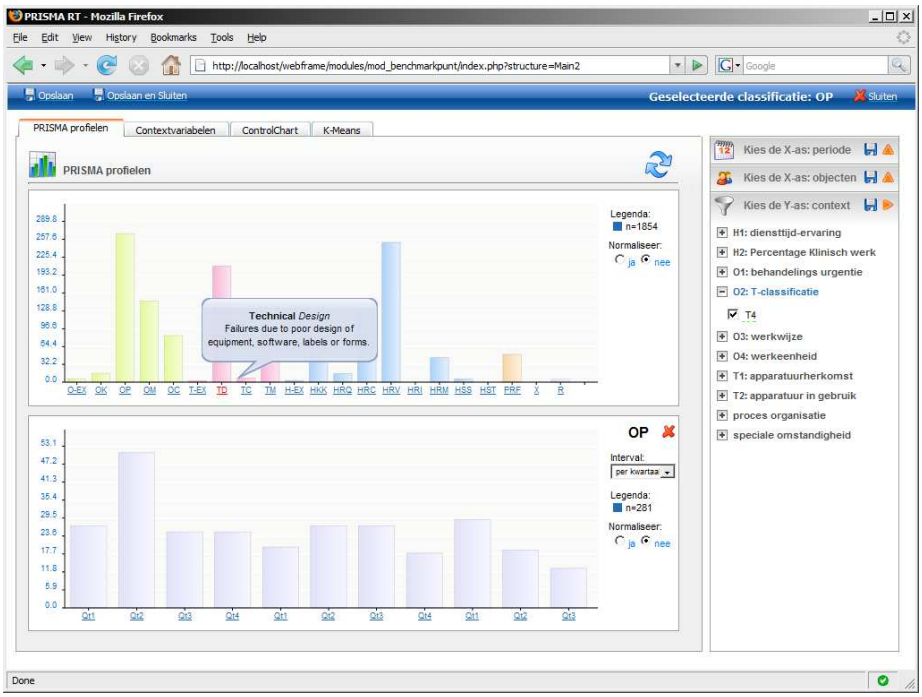
B.4. Conclusie C2 architectuur

De “lite” implementatie van de C2 Architectural style heeft ongeveer drie weken in beslag genomen. Naast de beloofde voordelen zijn er ook een aantal nadelen aan verbonden. Debuggen wordt lastiger (vanwege de vele requests die elkaar opvolgen), als tegenmaatregel is de functie toegevoegd dat de laatste honderd states worden bewaard en vanuit een lijst opnieuw gesimuleerd kunnen worden (met dezelfde context informatie als het originele request). Een bepaald component kan bij ontvangst van een message meer dan één vervolgoperaties uitvoeren, e.g. twee functies van het component hebben een message nodig en wat het component aan het uitvoeren is bepaald bij ontvangst welke functie bijvoorbeeld aangeroepen moet worden. Om een hoop if-statements in de code te vermijden is een toevoeging gemaakt om berichten te koppelen aan sequence-functies (bij het versturen wordt een sequence gekoppeld aan de message, deze bepaald *i.e.* volgens welk pad en functies de response worden afgehandeld) [10].

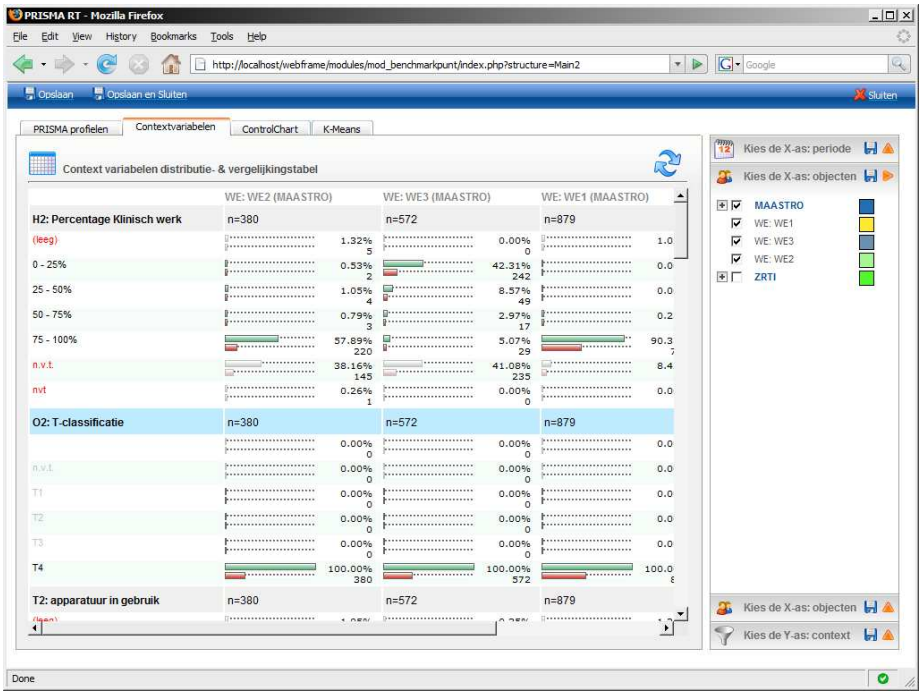
B.5. Prototype implementatie

Twee screenshots uit het prototype m.b.t. de huidige werkwijze zijn te vinden in figuren B.4 en B.5.

[10] Meest voor de hand liggende alternatief waar niet voor gekozen is: de hantering van één dergelijke sequence/“flow” per component.



Figuur B.4: Screenshot prototype PRISMAView.



Figuur B.5: Screenshot prototype ContextCompareView.

C Draaiboek gestructureerde analysevoering

#	View	Observatie	Verklaring analist.
1	PRISMA	1.1: In periode 2005 t/m juni 2007 is *Qr2 uit 2005* erg hoog	WE3 overgegaan naar Siemens apparatuur
1.1.1	PRISMA	Van de werkeenheden schiet *WE3* uit	WE3 overgegaan naar Siemens apparatuur
1.1.2	ContextCompare	In periode 2005 is proces organisatie voor 21.5% van de gevallen *Planning*.	Controles waren er toen (nog) niet.
1.1.3	TableView	Basisoorzaken omschrijvingen grotendeels: "geen controle (..)".	Volgens verwachting: ja Conclusie: Bevestigd controles ontbraken. Vervolganalyse: niet nodig. Vervolgtraject: niet nodig.
1	PRISMA	1.2.: In periode 2005 t/m juni 2007 is *Qr4 uit 2006* erg hoog voor werkeenheid WE1.	Zomer 2006 WE1 verhuisd voorbereiding van Heerlen naar Maastricht en Uitvoering bleef in Heerlen. Alleen bestraling werd nog in Heerlen gedaan. Eerste kwartaal 2007 is WE1 verhuisd. Volgens verwachting: ja Conclusie: Uitschieter had ook te maken met die verhuizing de daling die in de kwartalen van 2007 volgt bevestigd dit. Vervolganalyse: geen Vervolgtraject: geen
1	PRISMA	1.3.: In WE1 stijgt OP voor Linac en daalt voor Planning, maar niet erg drastisch.	Volgens verwachting: n.v.t. Conclusie: geen Vervolganalyse: geen Vervolgtraject: geen
2	PRISMA	2.1.: In WE1 is Epid zeer hoog in *Qr3 uit 2006*.	Epid monitoring?
2.1.1	TableView	De omschrijvingen lijken allemaal omtrent hetzelfde te gaan.	Meldgedrag is rond die periode gestimuleerd. Om actie te ondernemen en hopelijk het probleem te laten dalen. Volgens verwachting: ja Conclusie: Verklaring is meldgedrag. Vervolganalyse: geen

Figuur C.1: Experiment resultaten log (1/3)

			Vervolgtraject: geen
3	PRISMA	3.1.: In periode 2005 t/m juni 2007 is WE2 nauwelijks te vinden.	Dit is niet zoals de analist zich herinnert. In het verleden is er namelijk wel naar gekeken.
3.1.1	PRISMA	Alle overige Technische classificaties bekeken, ook nauwelijks data te vinden.	Volgens verwachting: nee Vervolganalyse: Excel sheets erbij gepakt (originele input voor het prototype). Data bevindt zich hier ook niet in. Vervolgtraject: Zal nader onderzocht worden. Conclusie: geen.
4	PRISMA	OM classificatie vertoond *zelfde patroon* als eerder gezien. WE1 eerst laag met oude werkwijze, daarna stijging.	Conclusie: 2006 zomer WE2 helemaal verhuisd alleen voorbereiding voor alle werkeenheden nog in Maastricht. Alleen WE1 gebeurde het bestralen nog in Heerlen. Voor WE1 in Qt1 van 2007 niet meer. Volgens verwachting: ja Vervolganalyse: geen Vervolgtraject: geen
5	PRISMA	OC classificatie vertoond ook *zelfde patroon*.	Conclusie: afgezien van het verhuizing verhaal valt niets op. Volgens verwachting: ja Vervolganalyse: geen Vervolgtraject: geen
6	PRISMA	TD classificatie *Qt4 uit 2006* voor WE2 wel erg hoog.	Specifiek proces? Vermoeden Epid proces.
	ContextCompare	Epid proces valt niets in op.	Misschien perongeluk onder Linac gescoort? Analist herinnert problemen met beeldkwaliteit.
	ContextCompare	Linac proces bevat de uitschieters	
	TableView	Linac basisoorzaak omschrijvingen allemaal: *problemen met de koppeling*.	Conclusie: bevestigd wat de analist al dacht Volgens verwachting: ja Vervolganalyse: geen Vervolgtraject: zorgen dat Epid niet meer onder Linac

Figuur C.2: Experiment resultaten log (2/3)

			gescoort wordt
7	PRISMA	7.1.: HRV classificatie begin 2007 erg hoog voor WE1.	Significant?
7.1.1	ControlCharts	Voor WE1 op maandniveau geen interessante uitschieters in het algemeen.	Misschien op technisch falen?
7.1.2	ControlCharts	Technisch falen significante uitschieter april 2007.	Dat is precies het moment dat WE1 in Maastricht arriveerde. Alleen op technisch falen uitschieter? Conclusie: oorzaak achterhaald Volgens verwachting: ja Vervolganalyse: geen Vervolgtraject: geen
7.1.3	ControlCharts	Organisatorisch niet, Menselijk bijna op hetzelfde moment.	WE2?
7.1.4	ControlCharts	WE2 ~Qt3 2005 menselijk falen hoog	Analist: te ver weg, even laten zitten.
7.1.5	ControlCharts	WE2 eind 2006, begin 2007 (vanaf oktober) organisatorisch hoog.	Problemen Epid ook na verhuizing.
7.1.6	ControlCharts	WE2 zelfde periode vanaf oktober technisch hoog.	Idem.
7.1.7	ControlCharts	WE3 februari 2007 Menselijk falen uitschieter.	Wat was er precies in februari aan de hand?
7.1.8	PRISMA	HRV schiet uit van het Menselijk falen.	Welk proces?
7.1.9	ContextCompare	Inplanning/afspraakbureau	Analist krijgt vermoeden waar het aan ligt, basisoorzaak omschrijvingen bekijken of dit het bevestigd.
	TableView	De basisoorzaken vallen onder de dokterassistentie en bevat zaken waar de werkeenheden niets mee kunnen.	Analist is op de hoogte van het probleem, doktersassistentie (DA) zal een aparte werkeenheid moeten worden zodat zaken niet onterecht onder de verkeerde werkeenheid worden gescoort. Conclusie: uitschieter is verklaard, en de DA had destijds te maken met een extra stimulans om te melden dus beïnvloed meldgedrag. Volgens verwachting: ja Vervolganalyse: geen Vervolgtraject: in de toekomst aparte werkeenheid DA.
8	K-Means	Classificatie OP, WE1, clustering op	Hier kunnen we bijvoorbeeld niets mee om de volgende

Figuur C.3: Experiment resultaten log (3/3)

1.	#1.1
probleem	2005 Q2 is erg hoog in de periode 2005 t/m juni 2007.
hypothese	WE3 is overgegaan naar Siemens apperatuur
voorspelling	Werkeenheden naast elkaar moet WE3 in die periode uitschieten.
experiment	in PRISMA view de drie werkeenheden naast elkaar plaatsen.
observatie	WE3 schiet uit t.o.v. WE1 en WE2 op dat moment
conclusie	confirmed
vervolgtraject	geen
2.	#1.1.2
probleem	2005 is in 21.5 % van de gevallen sprake van proces organisatie planning.
hypothese	controles waren er toen (nog) niet
voorspelling	terug te vinden in basisoorzaakomschrijvingen
experiment	in prototype TableView openen
observatie	basisoorzaak omschrijvingen grotendeels "geen controle"
conclusie	confirmed
vervolgtraject	geen
3.	#1.2
probleem	2006 Q4 is in de periode 2005 t/m juni 2007 erg hoog voor WE1
hypothese	Zomer 2006 is WE1 verhuisd: voorbereiding van Heerlen naar Maastricht, uitvoering bleef in Heerlen.
voorspelling	datum klopt
experiment	voorleggen domeinspecialist
observatie	domeinspecialist zoekt het op
conclusie	accepted
vervolgtraject	geen
4.	#1.3
probleem	WE1 stijgt OP voor Linac en daalt voor planning
vervolgtraject	niet drastisch genoeg
5.	#2
probleem	2006 Q3 epid zeer hoog WE1
hypothese	Epid monitoring probleem?
voorspelling	terug te vinden in basisoorzaakomschrijvingen
experiment	in prototype TableView openen
observatie	omschrijvingen gaan om hetzelfde, maar niet monitoring
conclusie	geen monitoring probleem
vervolgtraject	nieuwe hypothese
hypothese'	meldgedrag in die periode gestimuleerd
voorspelling	memo verstuurd in die periode met herinnering
experiment	domeinspecialist zoekt het op
observatie	meldgedrag is gestimuleerd vandaar extra veel meldingen
conclusie	confirmed
vervolgtraject	geen
6.	#3
probleem	In periode 2005 t/m juni 2007 is WE2 nauwelijks te vinden
hypothese	dit klopt niet moet meer informatie geregistreerd zijn.
voorspelling	technisch falen verwacht de analist dat er toch redelijk wat data van moet zijn
experiment	excel bestand openen sorteren op werkeenheid en count uitvoeren
observatie	excel bestand geeft zelfde resultaat als in prototype
conclusie	rejected
vervolgtraject	domeinspecialist zal hier nog naar kijken **

(** gaat om oude data, waar veel analyse over gedaan is en de prioriteit is daarom niet super hoog)

7.	#4
probleem	OM classificatie vertoond zelfde patroon als eerder over periode 2005 t/m juni 2007.
hypothese	komt overeen met verhuizing WE2
voorspelling	stijging en dalingen komen overeen met Q3,4 2006 en Q1 van 2007 (verhuizingen)
experiment	in prototype kijken of dit het geval is
observatie	stijging en dalingen komen overeen
conclusie	confirmed
vervolgtraject	geen
8.	#5
probleem	OC classificatie vertoond zelfde patroon als eerder over periode 2005 t/m juni 2007
hypothese	afgezien van het herhalende verhuizings-patroon vallen er meer zaken op
voorspelling	meer problemen die structureel lijken terug te komen
experiment	in contextcomparetable kijken en tableview op basisoorzaakomschrijvingen.
observatie	alle karakteristieken worden herkend als problemen m.b.t. verhuizing (zijn inmiddels opgelost)
conclusie	rejected
vervolgtraject	geen
9.	#6
probleem	2006 Q4 TD classificatie voor WE2 wel erg hoog.
hypothese	ligt het aan Epid proces (problemen van bekend m.b.t. beeldkwaliteit -analist)
voorspelling	Linac proces zou uit moeten schieten t.o.v. andere processen in de selectie
experiment	in prototype ContextCompare bekijken en TableView
observatie	Linac proces is dominerend, en binnen die selectie komen omschrijvingen neer op "problemen met de koppeling"
conclusie	confirmed
vervolgtraject	geen
10.	#7
probleem	2007 Q1 HRV classificatie erg hoog voor WE1
hypothese	Ligt aan Technisch falen.
voorspelling	Technisch falen moet t.o.v. andere basisoorzaken uitschieten
experiment	in prototype Control Charts de typen falen op variatie onderzoeken
observatie	Technisch falen significante uitschieter in april 2007 (vlak na verhuizing WE1)
conclusie	hypothese herformulering
hypothese'	Ligt aan technisch falen ten gevolg van opstartproblemen WE1 in Maastricht
conclusie	confirmed
vervolgtraject	geen
11.	#7.1.3
probleem	2005 Q3 is voor WE2 menselijk falen erg hoog
hypothese	??
voorspelling	meer informatie te vinden in overige views
experiment	ContextCompareTable en TableView nageslagen
observatie	Analist kan geen conclusie trekken op basis va de informatie (ook te lang geleden)
conclusie	geen
vervolgtraject	geen (wordt niet meer relevant geacht omdat l zo lang geleden is)
12.	#7.1.5
probleem	2006 vanaf oktober tot begin 2007 organisatorisch falen voor WE2 erg hoog.
hypothese	Lag aan Epid proces
voorspelling	in basisoorzaken moeten opvallen
experiment	kijken in TableView van prototype
observatie	in prototype TableView bevestigd, domeinspecialist herinnert zich de problemen met dat proces
observatie	WE2 in zelfde periode vanaf oktober technisch ook hoog
conclusie	confirmed
vervolgtraject	geen
13.	#7.1.7
probleem	2007 februari is menselijk falen voor WE3 erg hoog
hypothese	een specifiekere classificatie is de oorzaak
voorspelling	in die periode is een specifieke classificatie de leidende oorzaak
experiment	in prototype PRISMA profiel bekeken voor die periode
observatie	HRV classificatie schiet uit
conclusie	verdere drill-down, het proces achterhalen
hypothese	een specifiek proces is de oorzaak
voorspelling	processen vergelijking moet aantonen welk proces
experiment	in prototype ContextCompare view
observatie	het proces dat eruit springt is inplanning/afsprakenbureau
hypothese'	analist vermoedt dat de dokterassistentie (DA) onterecht onder verkeerde werkeenheden scoort.
voorspelling	dat kan de analist bepalen op basis van de basisoorzaakomschrijvingen
experiment	prototype TableView naslaan
conclusie	confirmed
vervolgtraject	in de toekomst komt er een aparte werkeenhe DA

Figuur C.4: Experiment resultaten bewerkt.

D Goodness of fit testresultaten

Handmatige inspectie Voordat de goodness of fit tests waren uitgevoerd zijn in totaal ongeveer honderd histogrammen gegenereerd en handmatig geanalyseerd: Alle classificaties bij elkaar per dag, week en maand, vervolgens groepen classificaties: technisch, organisatorisch, etc., voor dezelfde intervallen. Hetzelfde is gedaan met alle individuele classificaties (en af en toe nog specifiekere selecties). Deze zijn allemaal uitgeprint, allemaal bekeken, en d.m.v. inductie is daar het volgende uitgekomen.

De verdelingen per dag en per week zien er over het algemeen goed uit, per maand is bijna altijd slechter. Per maand wordt de lambda meestal te hoog en vervolgens vallen metingen ver buiten de verdeling, met een hoop gaten in de histogrammen. Per dag is de lambda weer vrij laag en vallen bijna nooit metingen buiten de verdeling. Per week vallen wel metingen buiten de curve, wat ook te verwachten is omdat de stroom aan incidenten niet altijd even constant is (i.v.m. meldgedrag, ..).

Resultaten Chi-kwadraat goodness-of-fit test

selectie	interv.	lambda	df	Chi^2	Resultaat
alle classificaties	dag	1.73610	10	3.81149	ACCEPTED 0.95545 >0.05
	week	12.0070	27	14.26585	ACCEPTED 0.97852 >0.05
	maand	52.0303	19	120.55741	REJECTED 0.00000 <0.05
technische classificaties	dag	0.66431	7	1.02343	ACCEPTED 0.99444 >0.05
	week	4.65957	15	9.26143	ACCEPTED 0.86346 >0.05
	maand	19.9091	15	161.69979	REJECTED 0.00000 <0.05
technische classificatie TD	dag	0.49343	5	0.21126	ACCEPTED 0.99899 >0.05
	week	3.46099	11	1.28940	ACCEPTED 0.99982 >0.05
	maand	14.7879	16	188.00438	REJECTED 0.00000 <0.05

Tabel D.1: Resultaten Chi-2 goodness-of-fit test.

Kolmogorov-Smirnov goodness-of-fit test uitkomsten De uitkomsten van de goodness-of-fit tests op alle-, groepen van- en individuele classificaties zijn te vinden in de tabellen uit figuren D.1 en D.2.

	N	Poisson Parameter(a,b)	Most Extreme Differences			Kolmogorov-Smirnov Z	Asymp. Sig. (2-tailed)
		Mean	Absolute	Positive	Negative		
alles	143	12,0070	0,225	0,225	-0,121	2,696	0,000
organisatorisch	140	8,0714	0,158	0,158	-0,099	1,870	0,002
technisch	141	4,6596	0,113	0,113	-0,057	1,343	0,054
menselijk	143	8,2657	0,173	0,173	-0,104	2,073	0,000
overig	139	1,3022	0,043	0,043	-0,022	0,507	0,959
O-EX	139	0,5036	0,072	0,072	-0,036	0,848	0,469
OK	139	0,5755	0,078	0,078	-0,037	0,918	0,368
OP	140	4,2786	0,098	0,098	-0,052	1,154	0,139
OM	139	3,6835	0,158	0,158	-0,109	1,860	0,002
OC	139	2,7194	0,165	0,165	-0,079	1,944	0,001
T-EX	139	0,1223	0,014	0,014	-0,007	0,170	1,000
TD	141	3,4610	0,087	0,087	-0,040	1,032	0,238
TC	139	0,1511	0,004	0,004	-0,004	0,048	1,000
TM	140	1,3214	0,140	0,140	-0,040	1,661	0,008

Figuur D.1: Kolmogorov-Smirnov goodness-of-fit resultaten tabel i/ii.

H-EX	139	0,1223	0,022	0,022	-0,015	0,255	1,000
HKK	139	0,8921	0,087	0,087	-0,054	1,021	0,248
HRQ	140	0,2429	0,023	0,023	-0,018	0,269	1,000
HRC	139	1,8417	0,072	0,072	-0,040	0,845	0,473
HRV	142	3,4648	0,114	0,114	-0,059	1,357	0,050
HRI	142	5,0493	0,203	0,203	-0,079	2,424	0,000
HRM	140	0,4857	0,085	0,085	-0,050	1,003	0,267
HSS	139	0,1799	0,021	0,021	-0,014	0,244	1,000
HST	139	0,0216	0,000	0,000	0,000	0,003	1,000
PRF	139	1,0504	0,053	0,053	-0,025	0,626	0,829
X	139	0,1007	0,002	0,002	-0,002	0,029	1,000
R	139	0,1007	0,009	0,009	-0,007	0,112	1,000
leeg	139	0,0935	0,003	0,003	-0,003	0,036	1,000

Figuur D.2: Kolmogorov-Smirnov goodness-of-fit resultaten tabel ii/ii.

E Control Charts

E.1. Interpretatie van de XmR chart

Verzamelde richtlijnen ter interpretatie van de x Chart, (hoogstwaarschijnlijk) is er sprake van een speciale oorzaak (lack of control) indien:

- Als een meting meer dan 3σ van de middenlijn valt. De kans hierop is heel klein en is dus onwaarschijnlijk door toeval ontstaan.
- Als 2 uit 3 opeenvolgende metingen aan dezelfde kant met meer dan 2σ afstand van de middenlijn vallen (derde punt mag aan beide kanten vallen).
- Als 4 uit 5 opeenvolgende metingen aan dezelfde kant met meer dan 1σ afstand van de middenlijn vallen (vijfde punt mag aan beide kanten vallen).
- Als 8 opeenvolgende waarden aan dezelfde kant vallen van de middenlijn.

Verzamelde richtlijnen ter interpretatie van de mR Chart, (hoogstwaarschijnlijk) is er sprake van een speciale oorzaak (lack of control) indien:

- Als een meting meer dan 3σ van de middenlijn valt. De kans hierop is heel klein en is dus onwaarschijnlijk door toeval ontstaan.

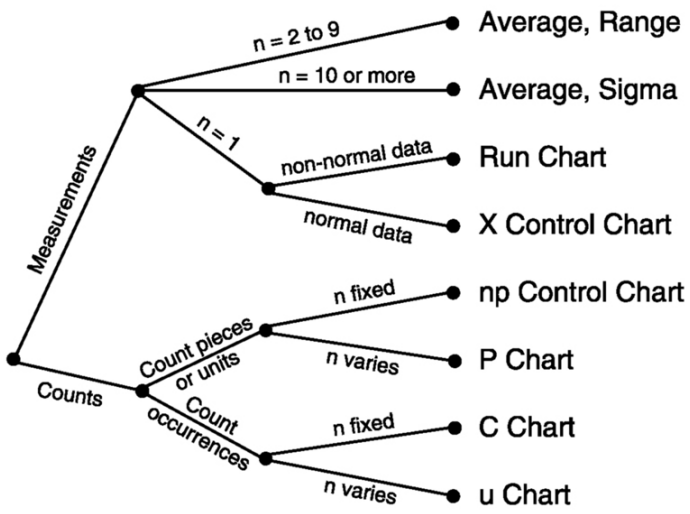
E.2. Keuze type Control Chart

Op basis van een tweede beslisboom (figuur E.3, uit [32, 31]) wordt de XmR (Individual X and Moving Range) chart geadviseerd; via de eerste beslisboom (figuur E.1) blijkt de p Chart echter de uitkomst. Echter hebben we geen goede noemer, zoals het aantal behandelingen of aantal mensen op de afdeling. De samplegrootte verschilt dus de grenzen zijn niet statisch, de p Chart kan dan moeilijk te interpreteren zijn[1] (de dynamische grenzen hoeven niet altijd nodig te zijn [23, p. 408]). De XmR chart wordt dus geadviseerd bij metingen met een samplegrootte van één, maar is volgens Pyzdek ook in ons geval (waarschijnlijk) geschikt [22]; “In other words, if all that are available are the percentages, the X chart provides an excellent approximation to the p chart. The same conclusion applies to data for the np chart, c chart, u chart, sigma chart, etc.”.

E.3. De XmR chart

Dit is een Control Chart waar de samplegrootte één is, daarom kunnen we geen range (R) uitrekenen over de verschillende *samples*. Wel een Moving Range (mR), dat is de variatie tussen twee of meer metingen. Gebruikelijk is het om twee metingen samen te nemen (e.g. op februari het verschil met

[1] “Analysis of p chart patterns between the control limits is extremely complicated if the sample size varies because the distribution of p varies with the sample size.”. (Pyzdek, [23, p. 406])



Figuur E.1: Decision tree for different control Chart types [23, p. 419].

januari, op maart het verschil met februari, etc., januari heeft op deze manier dus geen waarde). In het Individual X gedeelte zijn de individuele metingen geplot tussen grenzen gebaseerd op de middenlijn van de Moving Range grafiek.

Dit zijn de berekeningen voor de boven- en ondergrens (upper- en lower- control limits) van de individuele metingen (x Chart): $UCL_x = \overline{x} + (2.66)\overline{mR}$, $LCL_x = \overline{x} - (2.66)\overline{mR}$. Hier zijn $\overline{x}$ het gemiddelde van alle individuele metingen, 2.66 een Shewhart constante, $\overline{mR}$ het gemiddelde van alle individuele moving ranges. De grensberekeningen voor de grafiek: $UCL_{\overline{x}} = \overline{\overline{x}} + (2.66)\overline{mR}$, $LCL_{\overline{x}} = \overline{\overline{x}} - (2.66)\overline{mR}$. $\overline{\overline{x}}$ is het grote gemiddelde van alle subgroep gemiddelden, in dit geval gelijk aan aan $\overline{x}$. In deze grafiek is het gedeelte boven de middenlijn $\overline{\overline{x}}$ de *upper plot* en onder de *lower plot*. Bovengrens voor de moving range (mR Chart): $UCL_{mR} = (3.268)\overline{mR}$ (ondergrens is er niet). Hier is 3.268 weer een Shewhart constante en $\overline{mR}$ is het gemiddelde van alle individuele moving ranges. Een individueel moving range wordt zo berekend: $mR_i = |x_{i+1} - x_i|$, waar x een individuele meting voorstelt.

Als er een meting boven UCL_{mR} valt en 2/3e van individuele mR metingen valt beneden $\overline{mR}$, is er sprake van een *inflated* control limit (in het geval van categorische gegevens). Dan kan de grens aangepast worden naar: $UCL_{mR} = 3.144$ * *mediaan van moving range*.

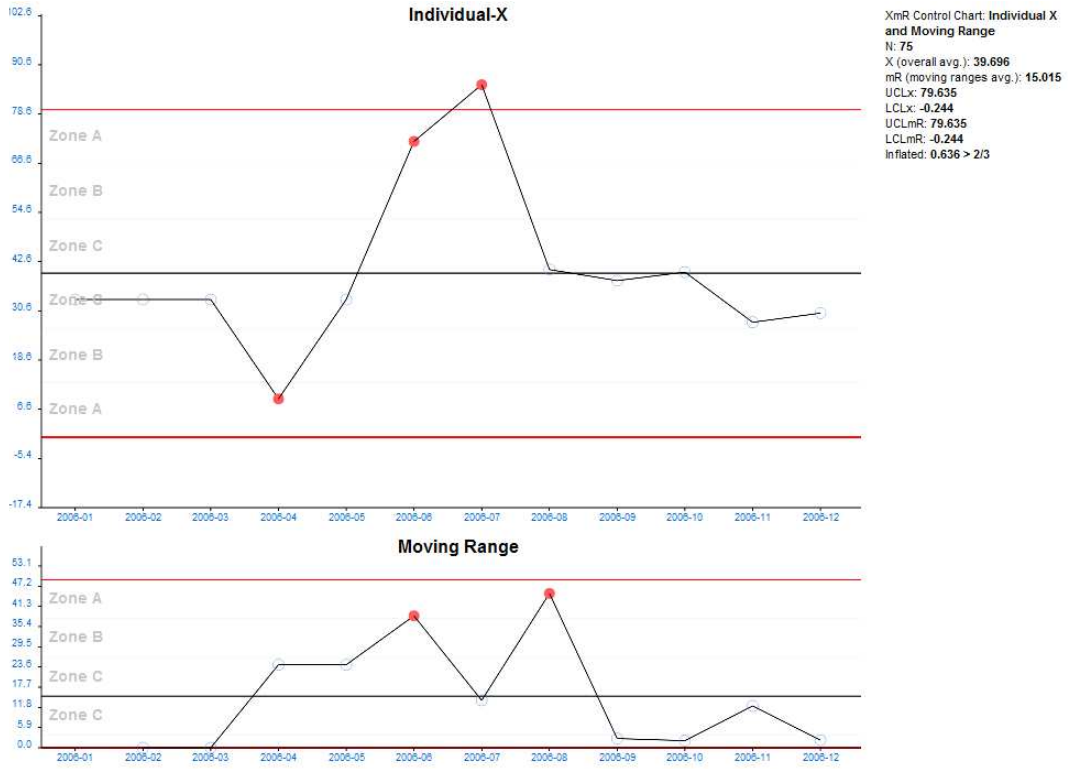
E.4. Normalisatie

Om toch enigszins te kunnen normaliseren is een keuze ingebouwd om als noemer een bredere selectie te hanteren. D.w.z. de selectie van "organisatorische falen" kan bijvoorbeeld uitgedrukt worden in "organisatorisch+technisch+menselijk+overig falen", en al het falen van een afdeling kan uitgedrukt worden in percentages van alle afdelingen van het betreffende instituut. Deze normalisatiekeuze is optioneel omdat het nooit is gegarandeerd dat deze *goed* normaliseert. Omdat de mogelijkheid bestaat dat uitschieters, de speciale oorzaken, (onterecht) gecompenseerd worden met dezelfde speciale oorzaken in de bredere selectie terwijl er dus wel wat aan de hand is.

In de normalisatie heerst dus de volgende grote aanname die luidt dat: "De specifieke faalwijze schiet uit ten opzichte van overige faalwijzen en de uitschieter vindt zijn weerslag niet of nauwelijks in de overige

faalwijzen.".

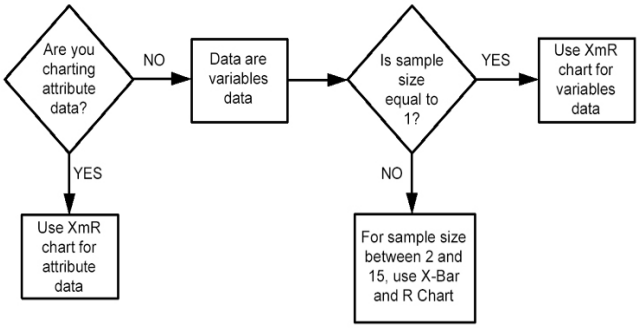
In figuur E.4 zijn de metingen (allen genormaliseerd naar 100%) te zien: type organisatorische, alle typen en de organisatorische in *percentage* van alle typen. Zo is te zien dat de organisatorische uitschieter rond augustus naar beneden gecorrigeerd wordt omdat deze in alle typen aanwezig is. Het Individual X gedeelte van de XmR chart geeft in ons geval dit percentage weer (of de indien niet genormaliseerd, de getelde faalwijzen). In de control chart gebaseerd op dit voorbeeld (figuur E.2) is ook alleen een speciale oorzaak te zien in de maand juli. Een control chart waar geen normalisatie gedaan wordt (figuur 6.3) laat iets heel anders zien, maar is betrouwbaarder omdat de aanname hierin niet gemaakt wordt.



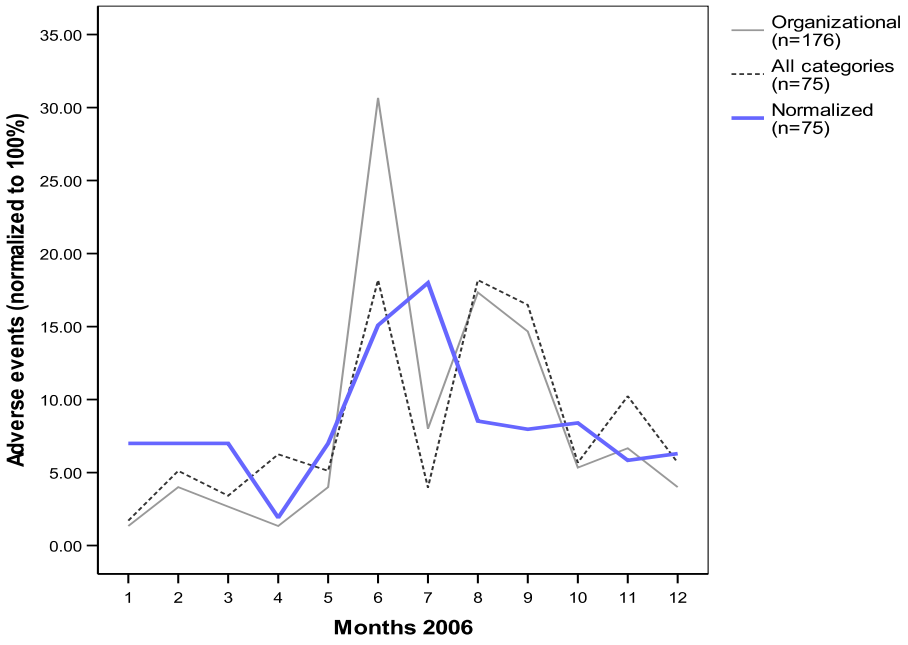
Figuur E.2: XmR MAASTRO 2006 Epid, organisatorische faalwijzen. Genormaliseerd.

Pyzdek zegt verder: "One could even argue that the simplicity of using a single chart instead of several charts outweighs the mathematical advantages in many cases.". Hierom en vanwege beschikbare tijd, is voorlopig alleen de XmR chart geïmplementeerd en niet beide. "In general, use the most simple method that leads to correct decisions.".

Conclusie De in deze sectie en in subsectie 6.3.1 gegeven voorbeelden laten zien dat de Control Charts (mits de data aan bepaalde eisen voldoet: niet te weinig nul waarden, *etc.*) ingezet kunnen worden om speciale- van procesinherente oorzaken te onderscheiden. Het prototype kan in de toekomst nog wel beter ondersteunen in het interpreteren van de metingen, dit is nog niet volledig geautomatiseerd. Zinnige toevoeging voor in de toekomst is het configureerbaar maken van de grenzen, het hanteren van grenzen die gebaseerd zijn op een extern instituut of groepen instituten. ■

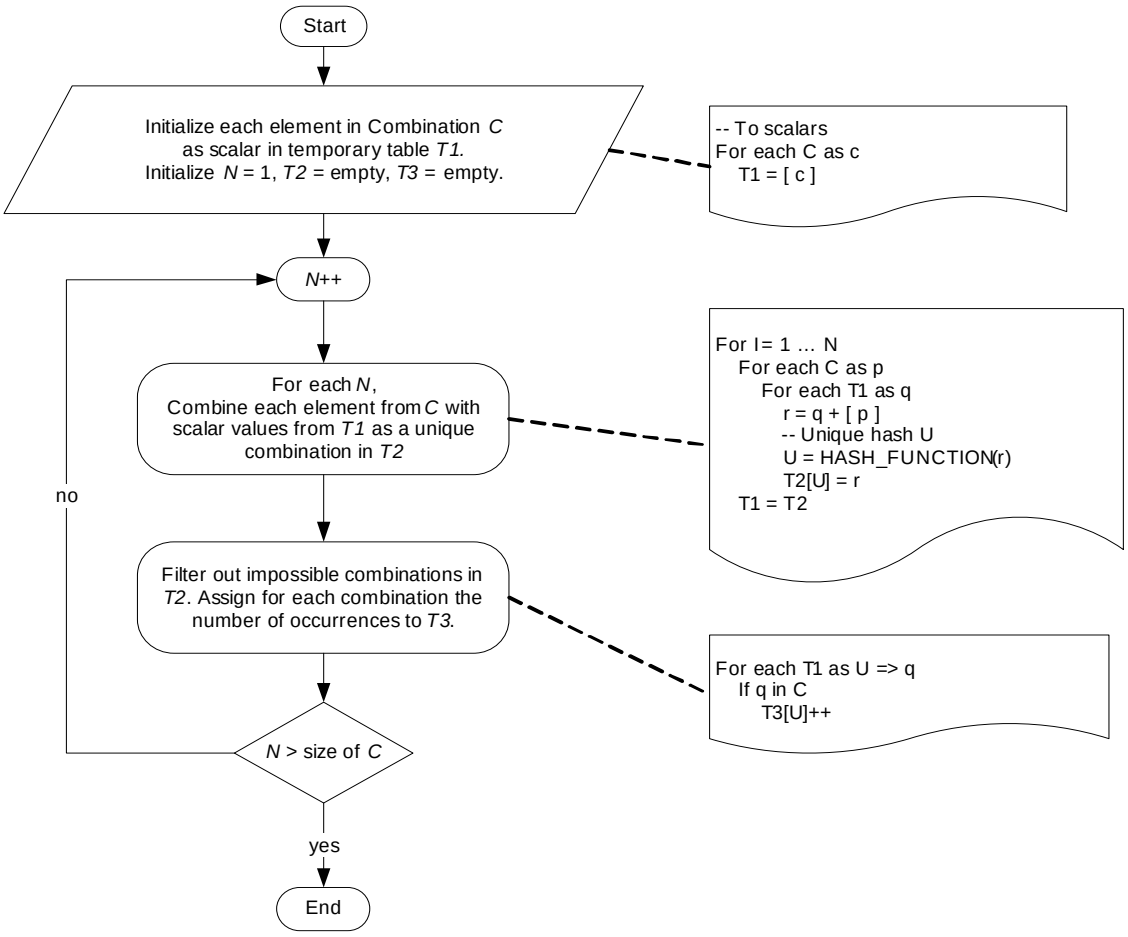


Figuur E.3: Decision tree for different control Chart types [31, p. 5].



Figuur E.4: MAASTRO afdeling ("proces") epid in 2006 faalwijzen als input voor de XmR Chart.

F Clusteringanalyse bijlagen



Figuur F.3: Schematische weergave “n-pair” algoritme.

Figuur F.1: MAASTRO proces inplanning/afspraakbureau 2006. k=4, n=177, clusters op classificatie. Weergave in clusters is *aantal voorgekomen (exacte) combinaties.*

Figuur F.2: MAASTRO volledige organisatie 2006 op classificatie OP. k=1, n=274, clusters op contextvariabelen en keuzen. Weergave in clusters is *aantal voorgekomen "paren" van combinaties (n-pair, n=5).*

F.1. Clustering op basis van Latent Semantic Indexing

De data geïmporteerd in Semantic Engine was over de periode vanaf januari 2005 t/m medio 2007 van de MAASTRO clinic.

Methode één: basisoorzaakomschrijvingen per analyse in één tekstbestand.

```
Cluster: aanwezig, bed, ontbreken, behandeld, locati
        files_meld/analyse748.txt
        files_meld/analyse163.txt
        files_meld/analyse119.txt
        files_meld/analyse116.txt
        files_meld/analyse306.txt
        files_meld/analyse1810.txt
        files_meld/analyse1378.txt
        files_meld/analyse175.txt
        files_meld/analyse168.txt
        files_meld/analyse109.txt
        files_meld/analyse753.txt
        files_meld/analyse1522.txt
        files_meld/analyse1249.txt
        files_meld/analyse763.txt
        files_meld/analyse766.txt
        files_meld/analyse894.txt
        files_meld/analyse765.txt
        files_meld/analyse416.txt
        files_meld/analyse160.txt
        files_meld/analyse1112.txt
        files_meld/analyse581.txt
        files_meld/analyse1200.txt
        files_meld/analyse1002.txt
        files_meld/analyse1576.txt
        files_meld/analyse1531.txt
        files_meld/analyse1369.txt

Cluster: zet, receptionist, zetten, brief, epid
        files_meld/analyse1566.txt
        files_meld/analyse1599.txt
        files_meld/analyse1688.txt
        files_meld/analyse935.txt
        files_meld/analyse2101.txt
        files_meld/analyse1429.txt
        files_meld/analyse1426.txt
        files_meld/analyse1600.txt

Cluster: recepti, regelen, wild, neer, passen
        files_meld/analyse1702.txt
        files_meld/analyse197.txt
...

Cluster: omtrent, fysica, dienst, vakj, middag
        files_meld/analyse1446.txt
        files_meld/analyse59.txt
```

```
...

Cluster: pijnmedicati, tijden, gemist
        files_meld/analyse1513.txt
        files_meld/analyse1514.txt

Cluster: azm, personeel
        files_meld/analyse1.txt
        files_meld/analyse310.txt

Cluster: bolus, gebruik
        files_meld/analyse891.txt
        files_meld/analyse202.txt
...
```

Methode twee: basisoorzaakomschrijvingen met gemaakte keuzes in contextvariabelen per analyse in één tekstbestand.

```
Cluster: aanwezig, nvt, box, meter, brief
        files_meld2/analyse_ext763.txt
        files_meld2/analyse_ext767.txt
...

Cluster: azm, overig, deuren, wachten, tevoren
        files_meld2/analyse_ext1.txt
        files_meld2/analyse_ext310.txt
...

Cluster: spo, virtuel, semi, sim, iso
        files_meld2/analyse_ext1513.txt
        files_meld2/analyse_ext351.txt
...

Cluster: ondersteun, patint, stuurt
        files_meld2/analyse_ext1971.txt
        files_meld2/analyse_ext1656.txt
...

Cluster: bolus, bestaand
        files_meld2/analyse_ext891.txt
        files_meld2/analyse_ext874.txt
...

Cluster: simul, veldcontrol
        files_meld2/analyse_ext119.txt
...
```

Methode drie: basisoorzaakomschrijvingen met gemaakte keuzes in contextvariabelen per oorzaak in één tekstbestand.

```
Cluster: apparatuur, emd, defect, meter, mlc
        files_meld3/oorzaak2663.txt
```

```
        files_meld3/oorzaak2179.txt
...

Cluster: registrati, digital
        files_meld3/oorzaak4705.txt
        files_meld3/oorzaak1198.txt

Cluster: kunnen, verplaatsen, mensen
        files_meld3/oorzaak852.txt
        files_meld3/oorzaak1714.txt
...

Cluster: benad, culturel
        files_meld3/oorzaak1905.txt
        files_meld3/oorzaak1918.txt
...

Cluster: epicor, altijd
        files_meld3/oorzaak4634.txt
        files_meld3/oorzaak4533.txt
...

Cluster: stereotacti
        files_meld3/oorzaak1810.txt
        files_meld3/oorzaak2222.txt
...

Cluster: team, terugkoppelen
        files_meld3/oorzaak2037.txt
        files_meld3/oorzaak859.txt
...
```