Nationaal Archief
Ministerie van Onderwijs, Cultuur en Wetenschap

Machine Learning en Automatische Classificatie

Versie 1.0

Datum: December 2018
Status: Definitief

Colofon

Projectnaam: E-Discovery voor Informatiemanagement
Projectleider: M. van Essen

Contactpersoon:
M. van Essen
T: +31 (0)6 55 26 78 77
F: +31 (0)70 331 54 77
E: mette.van.essen@nationaalarchief.nl
Postbus 90520 | 2509 LM Den Haag

Auteur: M. van Essen
Versie 1.0
Status: Definitief - vastgesteld 18 december 2018

Samenvatting

Ontwikkelingen op het gebied van kunstmatige intelligentie en machine learning gaan razendsnel. Er zijn steeds meer gegevens beschikbaar. Ook de toename van computerrekenkracht én de inzet van zelflerende algoritmen spelen een rol. Technologie kan organisaties en medewerkers ondersteunen om beter – en op een andere manier – om te gaan met grote hoeveelheden informatie. Tegelijk weten we dat we er met de inzet van technologie alleen niet zijn. Het zijn uitdagingen die vragen om een verandering binnen organisaties, het vakgebied en de informatievoorziening. Met het experiment Machine Learning en Automatische classificatie van e-mail hopen we een eerste stap te zetten in de richting van verandering. Een nieuwe manier van werken.

In de opzet van het experiment gaan we uit van de volgende aanname: *Als een spamfilter onderscheid kan maken tussen ongewenste mail (SPAM) en gewenste mail (HAM), dan kan een filter met machine learning ook onderscheid maken tussen twee andere klassen.*

Met deze aanname in het achterhoofd formuleerden we de volgende doelen:

- Het ontwikkelen van een classificatiemodel dat (ongelezen) binnenkomende e-mailberichten kan identificeren en toewijzen aan een bepaalde klasse;
- Het scheppen van vertrouwen en transparantie bij medewerkers in zelflerende systemen; de gebruiker traint het systeem zelf en ziet hierdoor direct het resultaat[1];
- Inzicht krijgen in de mogelijkheden en beperkingen van de verschillende zelflerende algoritmen die inzetbaar zijn voor een classificatieprobleem.

Er zijn veel onduidelijkheden en gevoeligheden rondom het gebruik van e-mailgegevens. Vaak gaat het om gevoelige informatie. Niet alleen persoonsgegevens, maar ook vertrouwelijke informatie in de e-mail-berichten zelf. Dit vraagt om maatregelen die datalekken en privacy-issues voorkomen. En eventuele risico's reduceren. Om dit goed te regelen, is juridisch advies ingewonnen en zijn uitgebreide maatregelen opgesteld (zie hfst. 2.2.1 Privacy en beveiliging).

Het experiment

Een belangrijke stap in het ontwikkelproces was het vaststellen van de klassen. We gebruiken als basis de toolkit 'Aanvulling op e-mailgedragslijn' van de Baseline Informatiehuishouding Rijksoverheid. We combineren deze met uitgangspunten van de nieuwe werkwijze voor e-mailarchivering van het programma Rijk aan Informatie (RaI, later gewijzigd in RDDI: Rijksprogramma voor Duurzaam Digitale Informatiehuishouding). We komen daarmee uit op de volgende twee klassen:

1. **Functionele** (of taak-gerelateerde) e-mailberichten: dit zijn berichten verzonden of ontvangen bij het uitvoeren van een taak.
2. **Ruis** (of niet taak-gerelateerde) e-mailberichten: dit zijn berichten die niet behoren tot de taak van medewerkers. Denk aan persoonlijke communicatie (privégebruik e-mail, communicatie tussen collega's onderling), dubbele informatie (cc-berichten, nieuwsbrieven, doorgestuurde berichten) en e-mail-berichten over afspraken, uitstapjes, traktaties, borrels enzovoorts.

Voor het ontwikkelen van het classificatiemodel gebruikten we supervised machine learning. Tijdens de ontwikkelperiode voedden we het systeem met ongeveer 3.500 voorbeelden van functionele berichten en e-mailberichten met ruis. Deze berichten zijn eerst handmatig gelabeld door de eigenaar. Door deze voorbeelden herkent het systeem eigenschappen van beide klassen. Het leert onderscheid te maken tussen functionele mails en mails met ruis. Het einde van de ontwikkelfase leverde een prototype op met een minimaal getraind classificatiemodel.

Vervolgens voedt de medewerker het systeem met eigen (van te voren geselecteerde), ongelabelde e-mail-berichten. Het maakt een voorspelling op basis van wat het in de ontwikkelfase leerde. Via een web-interface krijgt de medewerker een voorspelling van de klasse te zien (ruis of functioneel). Vervolgens geeft de medewerker aan of deze voorspelling correct is. Deze geeft deze informatie vervolgens terug aan het systeem. Het systeem leert daarna van de wijzigingen die de medewerker doorgeeft. Dit iteratieve proces herhaalt zich zolang er nieuwe e-mailberichten worden ingelezen.

Belangrijkste geleerde lessen

- **Maak privacy onderdeel van je experiment**
  We moeten nadenken over gegevensverwerking, vooral als een systeem dit gaat doen. Het organiseren van gesprekken, het opstellen van maatregelen en het bij elkaar brengen van de juiste personen kost tijd. De verwerking van (persoons)gegevens mag geen excuus zijn om een experiment niet uit te voeren.

- **Zelf ontwikkelen van een prototype helpt bij concretiseren van een probleem**
  Met dit experiment leerden we niet alleen hoe zelflerende systemen werken. We kregen ook beter inzicht in de problematiek die speelt rondom e-mail en ongestructureerde informatie in het algemeen.

- **De huidige infrastructuur, privacy maatregelen en de inrichting van organisatieprocessen brengt restricties voor het experimenteren met zich mee**
  Gebruik deze restricties als randvoorwaarden voor het uitvoeren van experimenten. Probeer tijdens het uitvoeren na te denken hoe het anders kan, ook in een ideale situatie. Plot dit op de echte situatie en kijk wat er in de toekomst haalbaar is.

- **Timeboxen en visualisaties helpen bij het maken van de juiste keuze**
  De beperkte ontwikkelperiode en de visualisatie van te maken keuzes, hielp ons in korte tijd een werkend prototype op te leveren. Dit droeg bij aan een geslaagd eindresultaat en maakte inzichtelijk wat er wel en niet kan.

Bevindingen experiment (samenvattend)

De feitelijke cijfers tonen aan dat er veel 'rommel' in onze mailboxen zit. Aan het begin van het experiment maakten we een inschatting dat waarschijnlijk 40 tot 50% van de e-mail weggegooid kan worden. Deze aanname klopte. Alleen al door ruis e-mail aan te merken en te verwijderen uit 'de grote bak' kunnen we tot bijna de helft aan opslag besparen. En dan hebben we het nog niet over de categorisering van functioneel belangrijk en functioneel onbelangrijk.

De afzonderlijke algoritmen gaven verschillende resultaten. Het algemene classificatiemodel werd gedurende de trainingsperiode 'slimmer'. Het werd goed in het herkennen van duidelijk ruis e-mailberichten en duidelijke functionele e-mailberichten. E-mails met zowel ruis als functionele boodschappen en twijfelgevallen werden moeilijker herkend. Dit is logisch. Medewerkers gaven aan dat zij het zelf ook lastig vonden om deze e-mailberichten te beoordelen.

Vertrouwen in een zelflerend systeem werkt anders dan we in eerste instantie dachten. Controle, of in elk geval het gevoel van controle hebben over het classificatiemodel, lijkt belangrijker dan inzicht hebben in het feitelijk functioneren van de algoritmen. Voor nu is de mogelijkheid om het algoritme te kunnen corrigeren nog een van de belangrijkste aspecten voor vertrouwen in het systeem.

Daar komt bij dat een medewerker verantwoordelijkheid voelt voor het goed functioneren van het systeem. Dat komt omdat deze zelf de trainer is. Je wilt het systeem immers zo goed mogelijk trainen. Door deze interactie wordt de eigen (menselijke) inconsequentie veel meer zichtbaar. En de medewerker wordt zich bewust dat een perfect systeem niet bestaat en niet nodig is. Het gevoel dat we het (nog) kunnen controleren is een sterke menselijke eigenschap. Die eigenschap moeten we niet onderkennen bij de acceptatie van deze systemen.

Vertrouwen in de persoon (of personen) achter het systeem draagt veel bij aan de acceptatie. Meer dan in eerste instantie was gedacht. Goede uitleg geven is nodig. Hoe werkt het systeem? Wat doe je ermee (welke vraag moet het beantwoorden)? En met welk doel (waarom zoeken we een antwoord op die vraag)? Het is zeer belangrijk dat je goed en nauwkeurig kunt uitleggen hoe je als organisatie omgaat met de gegevens van medewerkers.

Toepasbaarheid machine learning

Ondanks de beschikbaarheid van open source pakketten ondervonden we dat er behoorlijk wat werk nodig is om de gewenste resultaten te bereiken. Niet alleen voor het inzetten van de technologie van machine learning. Maar ook bij het uitvoeren van een experiment binnen de eigen organisatie. We moesten een weg vinden in een bijna oneindig aantal mogelijkheden aan oplossingen. Daarnaast bestaat een zelflerende systeem uit verschillende componenten die allemaal bijdragen aan het functioneren van het systeem. En aan de kwaliteit van de uiteindelijke voorspelling. Deze componenten moeten goed op elkaar afgestemd worden. En ze moeten passen in de infrastructuur van een organisatie.

Commerciële bedrijven bieden platforms aan die voor classificatiemogelijkheden gebruik maken van machine learning. Hierdoor ontstaat het beeld dat de technologie niet meer is dan een stukje software. Iets waar je een licentie voor koopt of wat je kunt installeren en dat dan direct werkt. Achter deze platforms draait een uitgebreide infrastructuur bij de bedrijven die het zelflerende component mogelijk maken. Hoe de gegevens precies verwerkt worden, welke algoritmen ze gebruiken en/of met welke features er getraind wordt is en blijft voor nu onduidelijk.

Ga je in zee met een commerciële partij? Verdiep je dan eerst in wat deze bedrijven echt aanbieden en wat jou als organisatie kan helpen. Kunstmatige Intelligentie en machine learning zijn buzz woorden. Het zijn technologische containerbegrippen die helpen een toepassing te verkopen. Laat je vooraf dus vooral goed informeren.

Inhoud

Inleiding

Ontwikkelingen op het gebied van kunstmatige intelligentie en machine learning gaan razendsnel. Dit is mogelijk door de beschikbaarheid van grote hoeveelheden gegevens. En door een toename aan computerrekenkracht én door de inzet van zelflerende algoritmen. Door deze ontwikkelingen borrelen nieuwe vragen op: kunnen zelflerende systemen bijdragen aan een betere informatiehuishouding? Is het mogelijk deze in te zetten voor het waarderen, selecteren en toegankelijk maken van ongestructureerde informatie binnen overheidsprocessen? Kunnen zelflerende algoritmen informatie identificeren en toewijzen aan een bepaalde klasse? Is de technologie al volwassen genoeg om ingezet te worden in een werkproces? En wat is nodig om vertrouwen te krijgen in de beslissingen die deze systemen voor ons maken?

Met het experiment Machine learning en Automatische Classificatie van e-mail ging het Nationaal Archief, in samenwerking met ICT Uitvoeringsorganisatie (ICTU), op zoek naar antwoorden op bovenstaande vragen.

Het experiment ontstond als onderdeel van het onderzoek E-Discovery voor Informatiemanagement[2]. Deze verkenning richtte zich op het inzetten van E-Discovery[3] methodiek. En op technologie voor de waardering (of classificatie) van ongestructureerde informatie binnen overheidsprocessen. Door dit zo vroeg mogelijk in het proces te organiseren wordt informatie beter toegankelijk. Deze is makkelijker te (her)gebruiken voor huidige en toekomstige doeleinden binnen een organisatie.

Binnen het vakgebied E-Discovery bestaan geavanceerde methoden die zelflerende systemen inzetten voor het identificeren, analyseren en classificeren van digitale informatie binnen grote gegevens verzamelingen. Dit noemen we het predictive coding proces. Dit proces traint het systeem aan de hand van beslissingen die de mens maakt. Een vorm van machine learning dus. Ons experiment kijkt of dit is in te zetten binnen het vakgebied van informatiemanagement.

De algemene doelen van het onderzoek E-Discovery voor Informatiemanagement dienden als uitgangspunt voor het experiment:

- Het ontwikkelen van nieuwe inzichten voor digitaal werken en informatiemanagement.
- Het opdoen en vergroten van kennis over de inzet van zelflerende systemen. Hoe kunnen deze systemen ingezet worden voor informatiemanagement? En, meer specifiek: voor het waarderen, selecteren en toegankelijk maken van informatie? Wat is het effect van deze systemen op de organisatie, de (werk) processen en de medewerkers?

Leeswijzer

Dit evaluatierapport gaat in op de resultaten van het vooronderzoek. Daarnaast komt de ontwikkeling van een prototype aan bod. Tenslotte is er aandacht voor het uitgevoerde experiment met een kleine groep medewerkers van het Nationaal Archief. De opzet van het rapport is uitgebreid om verschillende behoeften te bedienen.

Hfst.1: Theoretisch uitstapje ter verduidelijking van het begrip machine learning.

Hfst.2: De totstandkoming van het experiment. Opgebouwd uit verschillende onderdelen, het vooronderzoek, het ontwikkelen van een prototype in een lab-omgeving en het uitvoeren van een experiment met echte gegevens bij het Nationaal Archief. Knelpunten, keuzes en geleerde lessen worden meegenomen. Een samenvatting van de belangrijkste lessen staat in het laatste hoofdstuk.

Hfst.3: Gaat in op de resultaten van het daadwerkelijke experiment. Dit zijn zowel feitelijke resultaten als bevindingen aan de hand van interviews met de deelnemers van het experiment. Een samenvatting van de belangrijkste resultaten staat in het laatste hoofdstuk.

Hfst.4: Vertaalt de geleerde lessen naar algemene bevindingen over de toepasbaarheid van Machine Learning. We beantwoorden de vragen die we onszelf aan het begin van het experiment stellen.

Achtergrondinformatie
Bijlage A: Oplossingsrichting en onderzoeksvragen

1. Machine Learning: de basis

Machine learning is een onderwerp en een technologie waar veel over wordt gesproken. Het komt terug in de literatuur, in lezingen, in de krant, in tijdschriften en op vele blogs. Het wordt gepresenteerd als de heilige graal. Heb je gegevens? Dan lost machine learning al je problemen op. Veel organisaties houden de ontwikkelingen goed in de gaten. Zij zien het als een oplossing voor een diversiteit aan problemen zoals efficiëntievraagstukken, innovatie en het inzetten op verandering.

Wellicht kan machine learning dit allemaal tot stand brengen. Maar veel is gebaseerd op aannames en een eigen invulling. Dat de begrippen Artificial Intelligence (AI) en machine learning veel door elkaar worden gebruikt helpt hier niet bij. Beide begrippen hebben veel met elkaar te maken. Maar ze betekenen niet hetzelfde. In dit hoofdstuk volgt een korte toelichting op beide begrippen.

1.1 Artificial Intelligence & Machine Learning

Artificial Intelligence (AI) is het bredere concept dat machines en systemen steeds slimmer worden.

Het gaat om software of systemen die dingen kunnen die we over het algemeen als intelligente menselijke vaardigheden beschouwen (zoals analyseren, besluiten nemen en problemen oplossen).

Door het toepassen van machine learning is een AI in staat om van eerdere situaties te leren.

Als mens zijn we door de jaren heen gefascineerd geraakt door de mechanische mens. De manier waarop we tegenwoordig tegen AI aankijken is ontstaan met Alan Turing's publicatie Computing Machinery and Intelligence[4] uit 1950. In deze publicatie beschrijft hij de zogenoemde 'Turing-test'. Dat is een manier om de vraag te beantwoorden of machines kunnen denken. Deze vraag is volgens Turing zinloos. In plaats van vast te stellen of een machine kan denken vraagt hij zich af of een computer een spel kan winnen, The Imitation Game[5].

De afgelopen jaren gingen de ontwikkelingen rondom AI snel. Het meest tot de verbeelding sprekende voorbeeld van een AI is de zelfrijdende auto. Maar ook de persoonlijke assistent op een smartphone is een vorm van AI. Hoe meer tegen een smartphone gesproken wordt, des te beter deze de manier van praten leert te interpreteren. De assistent maakt steeds minder fouten en beantwoordt vragen steeds beter. De assistent leert de eigenaar van de smartphone steeds beter kennen. Het vermogen van de smartphone om te leren is een voorbeeld van een AI die gebruik maakt van Machine Learning technologie. Het vermogen van een systeem om te leren van ervaringen.

Net als AI is machine learning[6] geen nieuw vakgebied. Het wordt al jaren toegepast. Vooral bij statistische analyses. Door de enorme toename van gegevens en de vooruitgang in rekenvermogen van computers komen de ontwikkelingen ook binnen dit vakgebied in een stroomversnelling.

> ***Machine learning is het vermogen van computers of systemen om iets te doen dat vanzelfsprekend is voor mens en dier, namelijk het leren van ervaringen.***

Zelflerende algoritmen gebruiken wiskundige methoden om rechtstreeks van gegevens te leren. Dit doen ze zonder gebruik te maken van voorgeprogrammeerde regels en/of modellen. De algoritmen presteren beter als het volume van de beschikbare gegevens om van te leren toeneemt. Machine learning is een breed vakgebied. De meeste varianten hebben echter betrekking op het herkennen van patronen in gegevens.

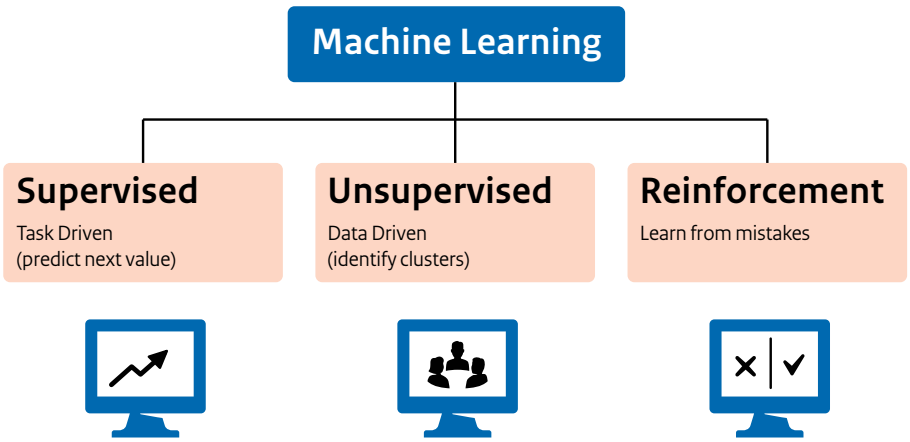
1.2 Soorten machine learning

Machine learning kent grofweg drie leermethoden. In dit project werkten we met supervised learning. Het trainen van de algoritmen gebeurt met gelabelde voorbeelden, input-informatie voor het systeem waarvan de gewenste uitkomst bekend is.

Een voorbeeld: Wordt een systeem gevoed met duizend afbeeldingen van een hond en duizend afbeeldingen van een kat? Dan leert het systeem verbanden te zien die horen bij het beeld van een hond en het beeld van een kat. Vervolgens kan een niet-gelabelde afbeelding (waar wel een hond of een kat op staat) aan het systeem worden gevoed. Het systeem geeft dan aan of het om een hond of een kat gaat.

Naast supervised learning bestaat er unsupervised learning. Hierbij heeft de input-informatie voor het leeralgoritme geen labels meegekregen. Het gaat om ongecontroleerd leren. Zonder sturing van voorbeelden met een gewenste uitkomst. Bij unsupervised learning ontdekt het algoritme op den duur zelf een structuur of patroon uit de input-informatie. Het clusteren van gelijksoortige informatie is een veel voorkomende toepassing van unsupervised machine learning.

In het experiment hebben we gesproken over het gebruik van unsupervised learning. Zo zou het prototype op basis van de gegevens voorstellen kunnen doen voor categorieën waarin e-mailberichten verdeeld kunnen worden. Maar gezien de korte tijdsperiode van de ontwikkelfase was dit niet haalbaar.



Figuur 1 - leermethoden machine learning

Een derde leermethode is reinforcement learning. Deze manier van leren wordt gezien als dé methode voor verdere ontwikkeling van AI. Bij reinforcement learning leert het systeem door interactie met de omgeving. Het systeem verzamelt trainingsvoorbeelden door het zogenoemde trail-and-error principe bij het uitvoeren van een bepaalde taak. Het systeem leert alles door zelf te proberen. Er zijn verschillende voorbeelden te vinden van algoritmen die zichzelf leren hoe zij een computerspelletje moeten spelen[7]. In het begin rommelt het programma maar wat aan. De ervaringen worden opgeslagen. De verwachtingspatronen worden bijgewerkt. Daarna probeert het programma het opnieuw.

1.3 Algoritmen

Een introductie in de wereld van machine learning kan niet zonder kort stil te staan bij het begrip algoritme. In de wiskunde en de computertechnologie is een algoritme een eenduidige beschrijving van hoe je een bepaald probleem oplost. Voorspellende modellen maken gebruik van algoritmen die zelflerend zijn. Deze algoritmen bepalen hoe een systeem de gegevens interpreteert. Ze bepalen de uitkomst van het leerproces. En daarmee de resultaten die je uiteindelijk krijgt. Algoritmen zijn overigens niet per definitie zelflerend. Automatische beslisregels in systemen kunnen ook gezien worden als algoritmen. Deze leren niet van input-informatie. Ze gebruiken criteria, opgesteld door mensen, om grote hoeveelheden gegevens te verwerken.

Een algoritme is een systematische set van activiteiten die je uitvoert op een gegevensset. Het is een procedure of een formule voor het oplossen van een (gegevens) probleem.

Een algoritme kun je zien als een container. Een container die een manier levert voor het opslaan van de methode die je gebruikt om een bepaald probleem op te lossen.

Welk algoritme je gaat gebruiken voor welk probleem is een belangrijke beslissing in het proces. Maar zeker niet de enige. Een inkijkje in wat er nog meer komt kijken bij het vormgeven van een zelflerend systeem is ergens anders te lezen in dit rapport.

2. Het experiment

Door de digitalisering van de maatschappij werken we anders en communiceren we anders met elkaar. De overheid is nog niet goed ingespeeld op de effecten van deze digitalisering.

Enkele van deze effecten zijn:

1. Het verwerken van gegevens en informatie gaat volgens analoge principes en methoden. Dit leidt tot fragmentarische opslag en beheer.
2. Informatie is niet meer uniek. Reeds gecreëerde en opgeslagen informatie verdwijnt snel uit het gezichtsveld.
3. Door toename van gegevens en informatie stijgen niet alleen de kosten voor opslag en beheer. Ook de risico's die een organisatie loopt nemen toe.
4. Het waarderen van informatie (het toekennen van waarde[8]) gebeurt achteraf. Dat brengt veel (handmatig) werk met zich mee.

E-mail is een informatiebron waarbij al deze effecten samenkomen. E-mail is nog steeds relevant. De overheid verstuurd en ontvangt grote hoeveelheden. Het aantal verzonden en ontvangen e-mailberichten binnen het Rijk is naar schatting minstens een miljard per jaar[9]. Het is de voornaamste vorm van communicatie. En een belangrijke bron van informatie voor de medewerkers, de organisatie en de informatievoorziening aan derden.

Binnen de overheid is de medewerker die e-mailberichten ontvangt of opmaakt zelf verantwoordelijk voor het beheer of het archiveren van deze berichten. Regels en werkwijzen zijn op meerdere manieren uit te leggen. En de uitvoering hiervan verschilt per organisatie en zelfs per persoon. Bewaaracties zijn vaak versnipperd en inconsequent. Losse e-mails worden in een Document Management System (DMS)[10] bij een dossier gevoegd.

De meeste berichten blijven toch achter in de mailbox van de desbetreffende medewerker. Maatregelen als automatische opschoning van mailservers en restricties op de omvang van de inbox zorgen dat medewerkers informatie op andere plekken en op andere manieren opslaan. E-mailberichten stappelen zich op. Ze vormen steeds grotere verzamelingen ontoegankelijke informatie in mailboxen, op servers en op netwerkschijven.

Technologie kan organisaties en medewerkers ondersteunen. Dit door beter en op een andere manier om te gaan met deze grote hoeveelheden aan informatie. Maar met de inzet van technologie alleen zijn we er nog niet. Het zijn uitdagingen die vragen om een verandering binnen organisaties, het vakgebied en de informatievoorziening. Met dit experiment hopen we een eerste stap te zetten in de richting van verandering en naar een nieuwe manier van werken.

2.1 Het vooronderzoek

Met het experiment wilden we inspelen op actuele ontwikkelingen binnen ons vakgebied. Welke praktijkvragen spelen er? En waar zit de behoefte tot verbetering? Waar en op welk gebied zijn er nieuwe inzichten nodig? En hoe kan technologie dit ondersteunen? Door het organiseren van verschillende expertmeetings, informatiebijeenkomsten en workshops, gingen we op zoek naar praktijkproblemen en uitdagingen die spelen binnen de (Rijks)overheid. Met de uitkomsten van deze bijeenkomsten zijn we aan de slag gegaan.

2.1.1 De oplossingsrichting

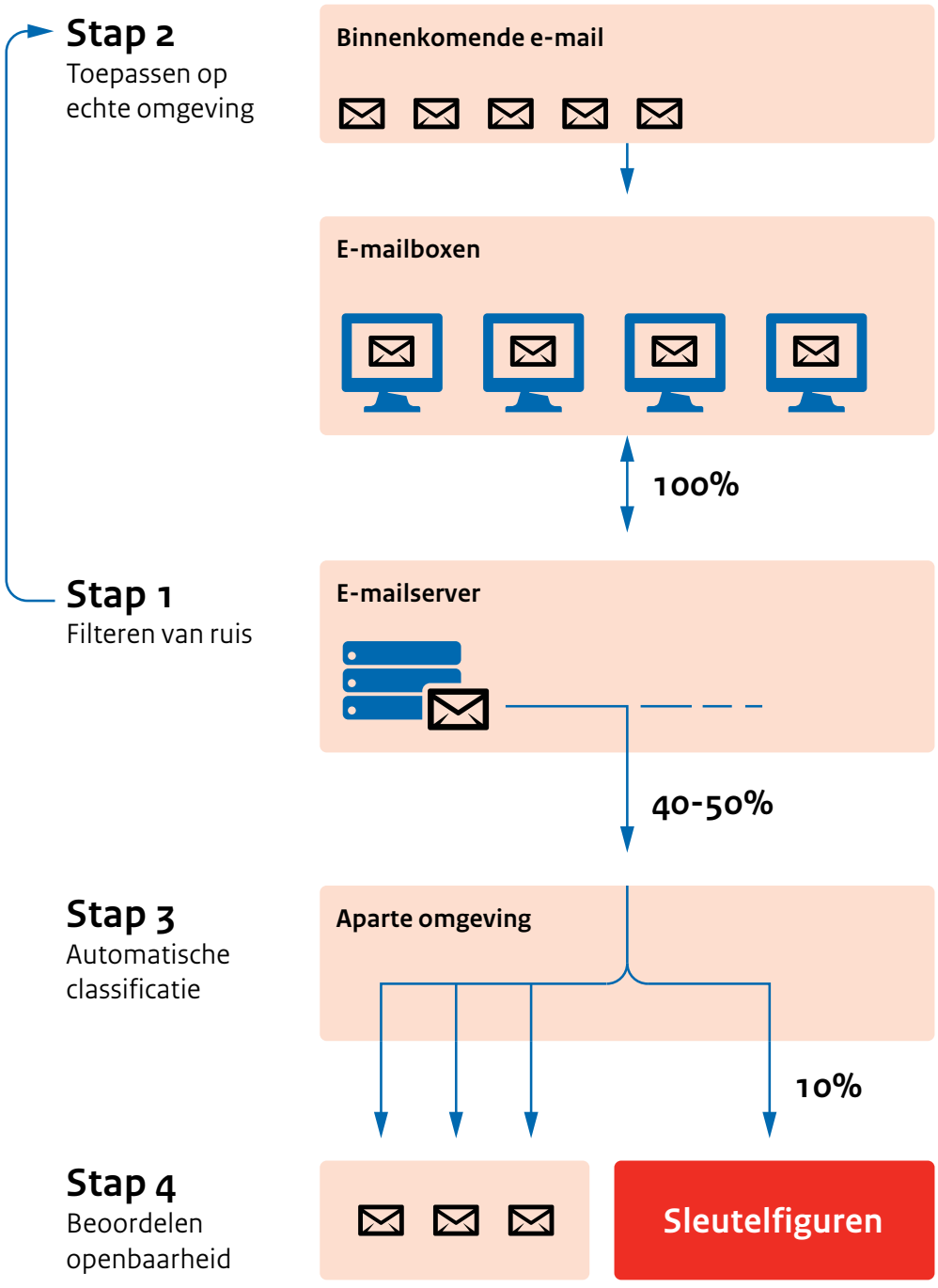
Doel: het ontwikkelen van een kader om binnen te experimenteren

De eerste contouren voor het experiment tekenden zich af tijdens een informatiebijeenkomst E-Discovery op 27 mei 2015. Het beeld dat bij veel organisaties e-mail niet of nauwelijks beheerd wordt, werd door de deelnemers bevestigd. Wat het eigenlijke probleem - de grote verzameling ontoegankelijke informatie - steeds groter maakt. Er werd gesproken over de behoefte om een gedeelte van het e-mailverkeer mee te laten lopen in de normale informatiestroom. Maar hoe zet je dit in gang? Hoe organiseer je projecten of experimenten als de toegang tot mailservers en mailboxen vaak moeilijk te regelen is vanwege privacy en wet- en regelgeving? In de discussie die hierop volgde werd die laatste vraag omgedraaid:

> *"Hoe kun je e-mail met technologie filteren zodat een persoon die de informatie beoordeelt alleen zakelijke e-mails of documenten voorgeschoteld krijgt?"*

Het gaat hier om het idee van het filteren van informatie. En het idee dat een systeem alle e-mailberichten mag of kan zien. Dus ook de gevoelige berichten. Vergelijk het met bodyscanners op vliegvelden. Toen deze net geïntroduceerd werden waren de lichamen van alle personen te zien. Inclusief vetrollen, buiken, billen en borsten. Dit leverde veel weerstand op. Het werd gezien als een inbreuk op de privacy. De oplossing: voor deze scanners werd een automatisch detectiealgoritme ontwikkeld. Het systeem scant het lichaam en ziet alles. Op het scherm - het filter - is alleen een standaardlichaam te zien. Ontdekt het systeem iets dat afwijkt? Dan geeft het standaardlichaam het gebied van de afwijking aan. De douaneambtenaar fouilleert de persoon op deze plek. Zo ondersteunt technologie de medewerker in het uitvoeren van zijn werk. Zonder inbreuk op de privacy.

Ongeveer in dezelfde periode kreeg het goed bewaren en toegankelijk maken van e-mailberichten politieke aandacht. Het programma 'Rijk aan Informatie' (RaI) ging van start met het project e-mailarchivering voor de Rijksoverheid[11]. Dit gebeurde in samenwerking met BZK/CIO Rijk. Met de nieuwe werkwijze voor het archiveren van e-mail[12], ontwikkeld binnen het project e-mailarchivering, vielen de puzzelstukjes voor dit experiment op zijn plaats. De ontwikkelde oplossingsrichting, het kader, combineert de nieuwe werkwijze voor e-mail met het idee van filteren van informatie.



Figuur 2 - Oplossingsrichting uit notitie E-Discovery voor Informatiemanagement (zie details Bijlage A)

We beperken ons hier tot uitleg van stap 1 en stap 2 van de oplossingsrichting. Dit omdat dit de stappen zijn die we hebben uitgevoerd. Een overzicht van het gehele kader met de daaraan gekoppelde onderzoeksvragen staat in Bijlage A.

In de nieuwe werkwijze krijgt de medewerker tien weken de tijd om persoonlijke en niet-relevante e-mailberichten te verwijderen. Of om deze te verplaatsen naar een aparte map in zijn of haar mailbox. Daarna worden alle overgebleven e-mailberichten naar een aparte omgeving verplaatst of gekopieerd[13]. We stelden onszelf de vraag of dit proces ondersteund kan worden door technologie. In de opzet van het experiment gaan we uit van de volgende aanname: Als een spamfilter onderscheid kan maken tussen ongewenste mail (SPAM) en gewenste mail (HAM), dan kan een filter met machine learning ook onderscheid maken tussen twee andere klassen.

Met deze aanname in het achterhoofd zijn de volgende doelen geformuleerd:

- Het ontwikkelen van een classificatiemodel dat (ongelezen) binnenkomende e-mailberichten kan identificeren en toewijzen aan een bepaalde klasse.
- Het scheppen van vertrouwen en transparantie bij medewerkers in zelflerende systemen. De gebruiker traint het systeem zelf en ziet hierdoor direct het resultaat[14].
- Inzicht krijgen in de mogelijkheden en beperkingen van de verschillende zelflerende algoritmen die inzetbaar zijn bij een classificatieprobleem.

Het vinden van partners bleek lastig

Het vinden van partners[15] om samen een experiment mee uit te voeren bleek lastiger dan van te voren bedacht. Het nut van het onderzoek was voor al onze gesprekspartners duidelijk. De opeenstapeling van ongestructureerde informatie buiten de beheersystemen is herkenbaar. Dit oplossen zonder de inzet van technologie lijkt onmogelijk. De vraag om tijd vrij te maken en mee te doen aan een experiment is van een andere orde. Daar komt bij dat de gekozen informatiebron weerstand opriep. Want mensen relateren e-mail direct aan het verwerken van persoonsgegevens. Dat klopt. E-mail raakt personen die e-mail als communicatiemiddel gebruiken. Maar ook de personen over wie de e-mailberichten gaan. Dit betekent dat privacyregelgeving[16] van toepassing is. Helaas hadden we tijdens de gesprekken nog onvoldoende zicht op de juridische consequenties hiervan. Dit bleek voor de meeste organisaties een belangrijke rede om niet mee te doen aan een experiment.

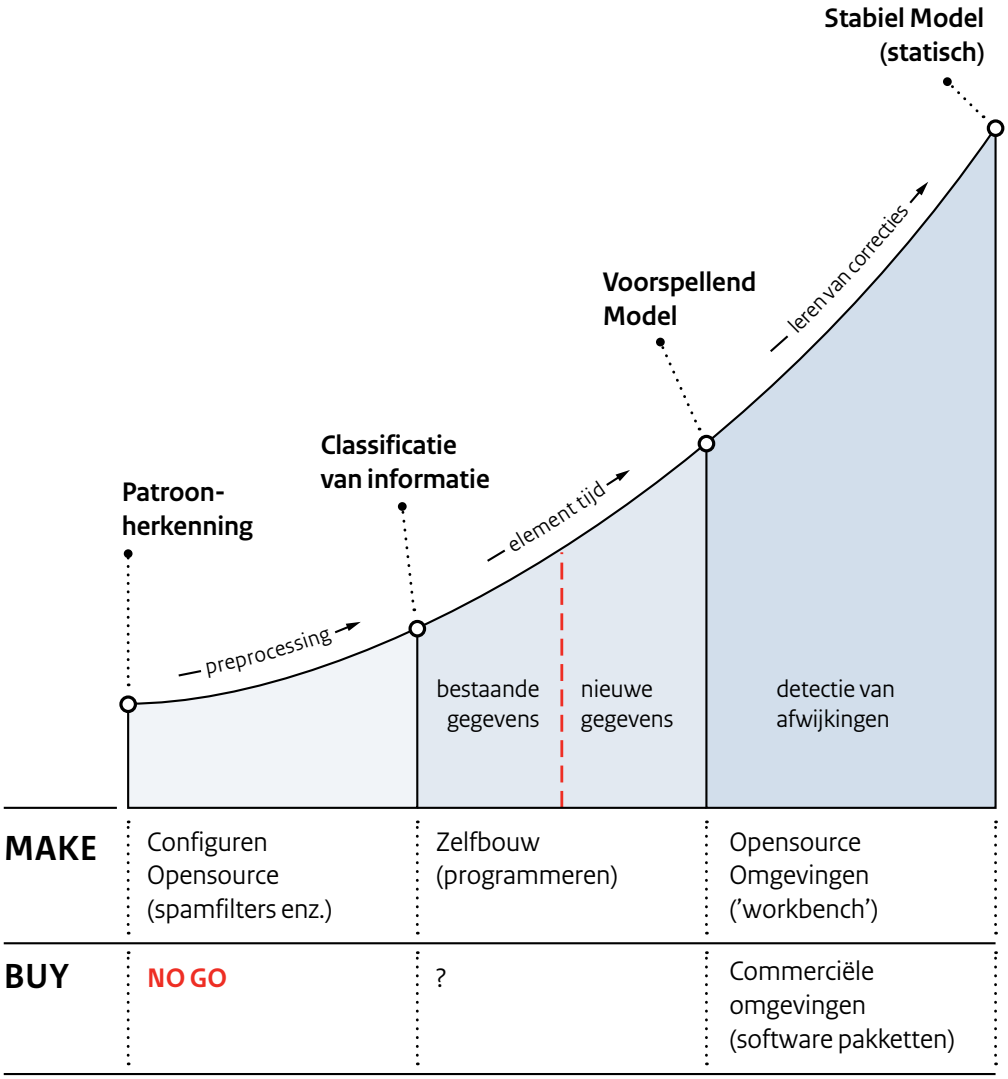
Om dit probleem op te lossen hebben we besloten een kleinschalig experiment uit te voeren bij het Nationaal Archief. Het Nationaal Archief is een overheidsorganisatie. Deze maakt gebruik van e-mail bij de dagelijkse werkzaamheden. Door zelf te ervaren hoe het is om een experiment met een nieuwe technologie uit te voeren krijg je inzicht in de problematiek. Zowel technisch, juridisch als organisatorisch. Als we zelf meer kennis hebben over het vraagstuk, dan kunnen we in de toekomst onze partners wel iets bieden. Privacyregelgeving en het verwerken van (persoons)gegevens zijn daarmee een belangrijk onderdeel van het experiment geworden.

2.1.2 Technologische verkenning

Doel: het vinden van een technologische oplossing voor het experiment.

Om de spamfilterhypothese te testen zijn we ons bij de technologische verkenning gaan richten op spamfiltertechnologie[17] en machine learning toepassingen voor classificatie. De eerste selectie van beschikbare en te overwegen technologieën gaf een groslijst[18] van ongeveer dertig mogelijkheden. Deze lijst maakt geen onderscheid in noodzakelijke randvoorwaarden, zoals expertise van mensen en vereisten aan de infrastructuur. Strikt genomen is het een alfabetische – niet uitputtende – lijst van onvergelijkbare grootheden. Van open source spamfilters tot softwarepakketten van commerciële partijen. En van machine learning platforms met oneindige mogelijkheden aan modellen en algoritmen.

Om grip te krijgen op deze hoeveelheid aan mogelijkheden ontwikkelden we onze eigen Machine learning leercurve. Deze leercurve begint met het voorzichtig proeven aan de technologie. Dit doen we door te kijken naar simpele mogelijkheden van patroonherkenning en classificatie[19]. De leercurve loopt op naar het ontwikkelen van een stabiel classificatiemodel met complexere algoritmen en leermethoden. Op deze curve hebben we onze groslijst geplot.



Figuur 3 - Machine learning 'leercurve'

Onduidelijke opslaglocaties en teveel functionaliteiten

We hadden als uitgangspunt dat de technologie zoveel mogelijk naar de gegevens toe werd gehaald. Dit betekende dat de gegevens waarmee we in het experiment aan de slag gingen – in dit geval e-mailberichten - niet of zo minimaal mogelijk buiten de technische omgeving van het Nationaal Archief werden opgeslagen en verwerkt werden. Cloudoplossingen en enkele technologiebedrijven vielen om deze reden direct af. Microsoft[20] kon ons bijvoorbeeld voor een experiment alleen een cloudoplossing aanbieden. Installatie van technologie bij het Nationaal Archief zelf was te kostbaar. Daar kwam bij dat grote partijen technologieën en oplossingen aanboden in de vorm van softwarepakketten of platforms. Dit betekende een legio aan mogelijkheden en functionaliteiten. Het testen van een 'enkele' functionaliteit, binaire classificatie, werd zonder het hele pakket aan te schaffen lastig. Het was alles of niets.

De eerste ervaring met open source spamfilters viel ook tegen. Het gebruik van deze filters was alleen mogelijk met cloud-based[21] e-mail. De e-mailgegevens werden vervolgens op een onbegrijpelijke manier en op onduidelijke servers verwerkt. Dit voldeed niet aan de privacyvoorwaarden. Zelf ontwikkelen leek daarom de meest voor de hand liggende oplossing.

Zelf ontwikkelen door prototyping
Bij het opstellen van de groslijst werd duidelijk dat er een ontzettend groot aanbod aan oplossingen, leermethoden en algoritmen is. Met een misschien nog grotere hoeveelheid aan partijen. Inzicht en praktische ervaring met machine learning was nodig om hierin een goede overweging te maken. Daarom hebben we besloten om in een lab-omgeving zelf een prototype te ontwikkelen. Is het prototype succesvol, dan installeren we dit binnen de Nationaal Archief omgeving voor een experiment. Lukt het ons niet, dan weten we dat we op een andere manier verder moeten. Het prototype en de technologie worden hiermee ingezet als vehikel om te leren. En niet om een probleem op te lossen.

Een andere reden om zelf te ontwikkelen was het opbouwen van transparantie en vertrouwen. We hebben hiermee de ontwikkeling en het trainen van het classificatiemodel in eigen hand. We weten welke algoritmen we gebruiken. En op welke kenmerken en met welke input we het model trainen. Bij alle partijen waarmee spraken was er een onduidelijk beeld van hoe de gegevens verwerkt werden. Hoe de classificaties tot stand kwamen en welke algoritmen er ingezet werden. Met andere woorden: we kregen black boxes aangeboden[22].

2.2 De ontwikkelfase

Vrij experimenteren doe je idealiter in een veilige omgeving. Bijvoorbeeld in een lab. Deze omgeving kun je zien als een speeltuin. Vervolgens wil je in een echte situatie met echte gegevens testen of je hypothese of je oorspronkelijke idee werkt. Buiten een geïsoleerde omgeving ontdek je andere dingen. De testomgeving die we binnen de infrastructuur van het Nationaal Archief realiseerden, moest zoveel mogelijk de 'echte situatie' nabootsen. Dit in tegenstelling tot de lab-omgeving wat een 'vrije speeltuin' was. Deze 'echte situatie' gaf ons aan het begin van de ontwikkelfase enkele restricties mee. Die restricties hadden invloed op de ontwikkeling van het prototype en de verschillende componenten.

In april 2017 lanceerde de ICTU het DIStributed Collaborative Information Platform, afgekort Discpl[23]. Deze lab-omgeving ondersteunde het Nationaal Archief bij het ontwikkelen van het prototype. Voor de ontwikkeling werd gekozen voor een agile aanpak. Hierdoor konden we relatief snel tot een prototype komen en stapsgewijs verbeteringen doorvoeren. We hadden de vrijheid om in een veilige omgeving te experimenteren met verschillende ideeën. Het aantal beschikbare uren van het team, de kosten en de doorlooptijd stonden vooraf vast. Na elke sprint[24] volgde een evaluatie en een analyse. Daarna werden de vervolgstappen bepaald.

2.2.1 Privacy en beveiliging

Doel: het in kaart brengen van de voorwaarden en een verdere invulling geven aan de maatregelen die risico's moeten beperken.

Zoals we zagen in hoofdstuk 2.1.1, waren er veel onduidelijkheden en gevoeligheden rondom het gebruik van e-mailgegevens. Werken met e-mailgegevens betekent werken met gevoelige gegevens. Dit gaat niet alleen om persoonsgegevens, maar ook om vertrouwelijke informatie in de e-mailberichten zelf. Dit vraagt om maatregelen die datalekken en privacy issues voorkomen en eventuele risico's reduceren. Om dit goed te regelen hebben we juridisch advies ingewonnen. We overlegden met beveiligingsfunctionarissen van het Nationaal Archief en de ICTU. Voor de ontwikkeling van het prototype en het uitvoeren van het experiment zijn de volgende uitwerkingen opgeleverd:

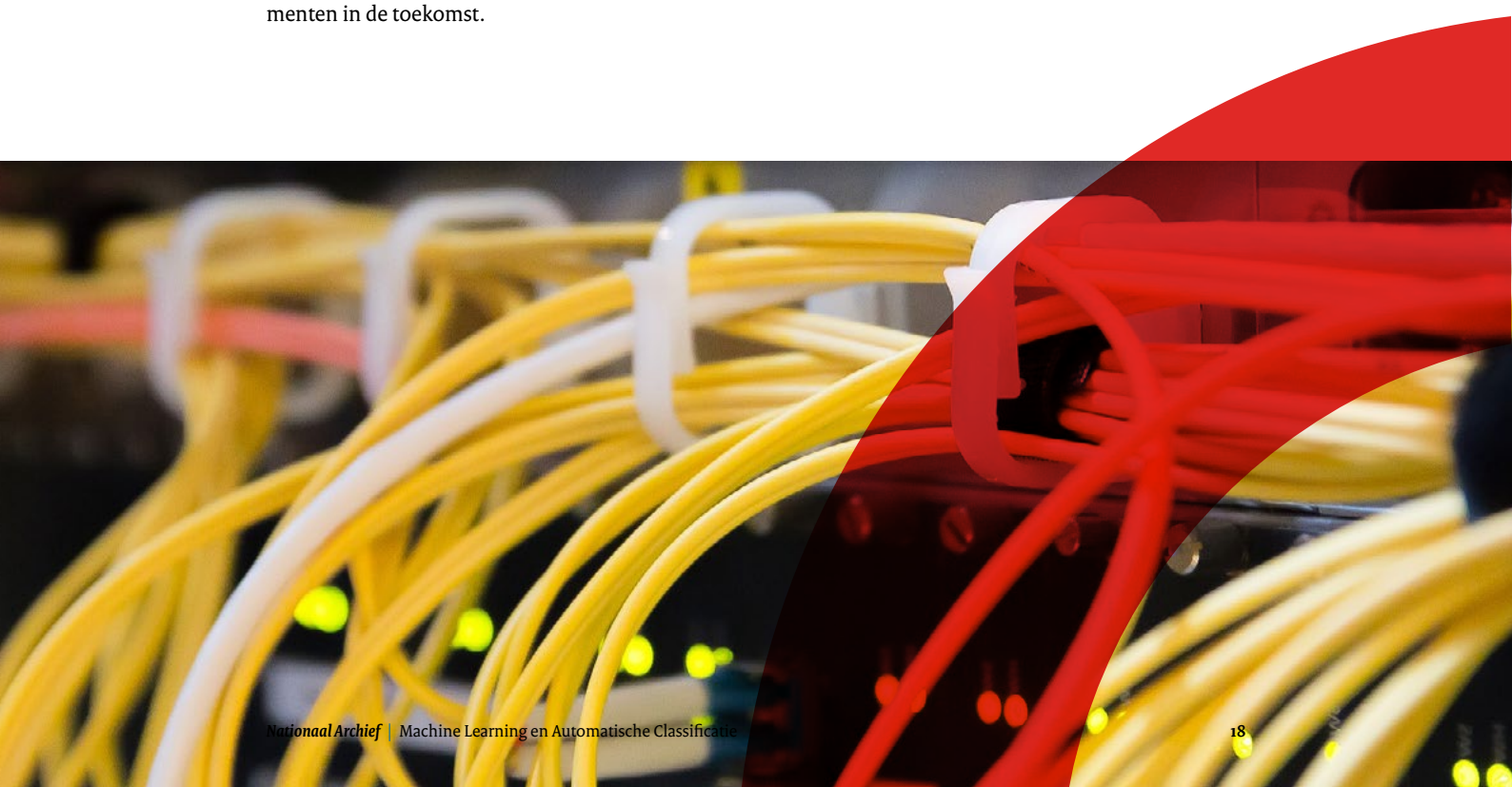
1. Een gebruikersovereenkomst tussen het Nationaal Archief en de ICTU waarin de georganiseerde maatregelen en juridische afspraken zijn vastgelegd.
2. Een overzicht van de maatregelen die we hebben getroffen. We stelden deze op door een quickscan risico-analyse uit te voeren.
3. Een afgestemde procedure tussen Nationaal Archief en ICTU voor datalekken[25].
4. Een informatieblad voor medewerkers van het Nationaal Archief over de genomen maatregelen en hun rechten bij deelname aan het experiment[26].

De genomen maatregelen bepaalden voor een deel de ontwikkeling van het prototype en de inrichting van het experiment. Dit zijn enkele belangrijke voorwaarden die invloed hadden op zowel het experiment als de ontwikkeling van het prototype:

- Deelname aan het experiment was op vrijwillige basis.
- We gebruikten kopieën van de e-mailberichten. Hierdoor had het experiment geen invloed op de dagelijkse werkstroom van de deelnemers. De hoeveelheid e-mailberichten nam niet toe en nam niet af.
- Aan het einde van het experiment hebben we de kopieën en alle naar personen herleidbare gegevens in één keer vernietigd.
- De e-mailberichten die nodig waren voor verdere training van het prototype, bleven binnen de werkomgeving van het Nationaal Archief.
- De deelnemers aan het experiment kregen enkel hun eigen e-mailberichten te zien.
- Het classificatiemodel maakte geen beslissing ten aanzien van de e-mailberichten, maar gaf een suggestie of voorstel aan de deelnemers.

Experimenteren tegenover gebruikersovereenkomst
Aan het begin van de ontwikkelfase was de opzet van het experiment onduidelijk. Ook hadden we het onvoldoende uitgekristalliseerd voor een gedegen juridisch advies. Dit kwam vooral door de gekozen ontwikkelvorm en de agile manier van werken. We wisten van tevoren niet wat drie maanden ontwikkelen zou opleveren. Dit staat haaks op de eisen die je in een gebruikersovereenkomst opstelt. Je geeft niet alleen aan waarom je de gegevens verwerkt, maar ook hoe je deze verwerkt. Een ding stond wel vast: voor de ontwikkeling van een classificatiemodel hadden we een trainingsset van gelabelde e-mailberichten nodig. Deze moesten we verplaatsen naar de lab-omgeving en zonder gebruikersovereenkomst konden we deze niet leveren.

Om de vaart erin te houden, hebben we de uitwerking van de overeenkomst gesplitst in twee onderdelen. Dit waren een overeenkomst voor de ontwikkeling van het prototype in de lab-omgeving en een overeenkomst voor het experiment met de medewerkers van het Nationaal Archief. De eerste overeenkomst was strenger door de noodzaak van het verplaatsen van e-mailberichten naar een andere omgeving. De verantwoordelijkheden lagen hierdoor niet alleen bij de eigen organisatie, maar ook bij een externe partner (de ICTU). De overeenkomst met de medewerkers was hierna relatief eenvoudig te realiseren. We konden putten uit de uitgebreide analyse, die we hadden uitgevoerd tijdens de ontwikkelfase. Deze analyse leverde een goede blauwdruk op voor het experiment bij het Nationaal Archief en voor eventueel andere experimenten in de toekomst.



2.2.2 Het prototype

Doel: Het ontwikkelen van een classificatiemodel wat (ongelezen) binnenkomende e-mailberichten kan identificeren en toewijzen aan een bepaalde klasse;

Een belangrijke stap in het ontwikkelproces was het vaststellen van de klassen. De toolkit 'Aanvullig op e-mailgedragslijn' van de Baseline Informatiehuishouding Rijksoverheid[27] gebruikten we als basis. Deze combineerden we met uitgangspunten van de nieuwe werkwijze voor e-mailarchivering. Zo kwamen we uit op de volgende twee klassen:

1. **Functionele** (of taakgerelateerde) e-mailberichten: berichten verzonden of ontvangen bij het uitvoeren van je taak.
2. **Ruis** (of niet taakgerelateerde) e-mailberichten: berichten die niet behoren tot de taak van de medewerkers. Denk aan persoonlijke communicatie (privégebruik e-mail, communicatie tussen collega's onderling), dubbele informatie (cc-berichten, nieuwsbrieven, doorgestuurde berichten) en e-mailberichten over afspraken, uitstapjes, traktaties, borrels enz.

Deze onderverdeling gaven we mee aan de deelnemers van het experiment. Het was een bewuste keuze om van tevoren zo min mogelijk gebruik te maken van vaste regels. We wilden het systeem de (intuïtieve) beslissing van de mens laten nabootsen. Het classificatiemodel bekeek enkel de inhoud (de body) van de e-mailberichten. Gestructureerde informatie van de headers, die veel 'slimme' e-mailtoepassingen analyseren, hebben we niet gebruikt. We waren niet op zoek naar een oplossing voor e-mailclassificatie specifiek, maar geïnteresseerd in de werking van tekstclassificatie in het algemeen. Als het werkt op de tekstuele inhoud van een e-mailbericht, dan werkt het, in theorie, ook op de andere tekstdocumenten.

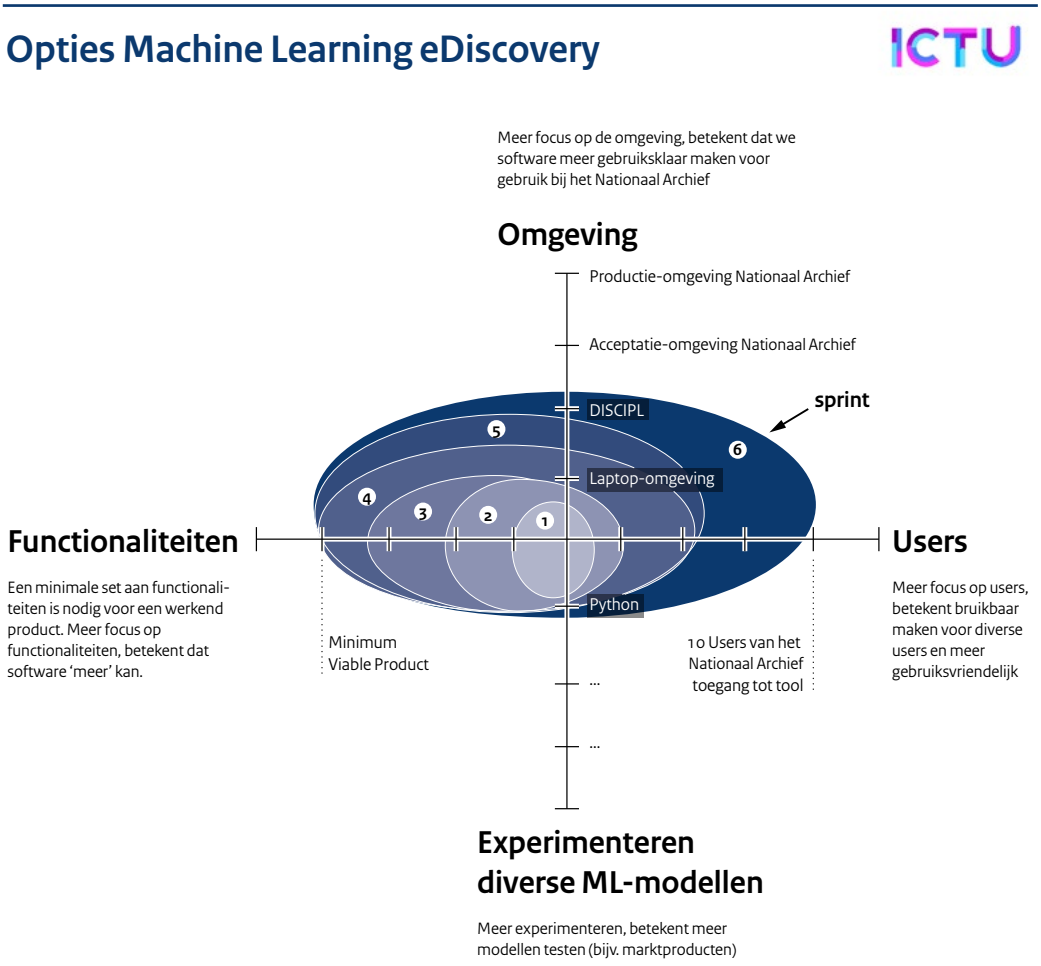
Voor de ontwikkeling van het classificatiemodel gebruikten we supervised learning, zoals beschreven in hoofdstuk 1. Tijdens de ontwikkelperiode voedden we het systeem met enkele duizenden (ongeveer 3.500) voorbeelden van functionele en ruis e-mailberichten. Deze berichten heeft de eigenaar ervan van tevoren handmatig gelabeld. Aan de hand van deze voorbeelden ging het systeemeigenschappen van beide klassen herkennen. Het leerde zichzelf onderscheid te maken tussen functionele en ruis mails. Aan het einde van de ontwikkelfase leverden we een prototype op met een minimaal getraind classificatiemodel.

Wat is het eindproduct?
De dunne scheidslijn tussen het opleveren van een prototype en het experimenteren met zelflerende systemen leverde af en toe communicatieproblemen op. De constructie van opdrachtgever (Nationaal Archief) en opdrachtnemer (ICTU), waarbinnen het prototype werd ontwikkeld, was daarbij lastig. Voor het Nationaal Archief lag het zwaartepunt op experimenteren. We wilden leren. Dit kon betekenen dat het uiteindelijk niets concreets opleverde, maar dat het nog steeds een geslaagd experiment was. In de traditionele rol van opdrachtnemer heb je het dan lastiger. De ICTU worstelde met de vraag: wanneer is het af en wanneer is het Nationaal Archief tevreden? Het opstellen van leerdoelen hielp om deze vragen te beantwoorden. Een af product, in dit geval een werkend prototype, blijft een duidelijker eindresultaat dan een document met geleerde lessen.

Bovendien bestond er bij de verschillende teamleden een eigen beeld van wat het eindresultaat kon of moest zijn. In de eerste sprints praatten we veel langs elkaar heen. Het korte tijdsbestek van drie maanden zorgde dat de juiste keuzen maken belangrijker was dan ooit. Een doorbraak daarin kwam toen we de keuzen zijn gaan visualiseren langs vier assen, namelijk:

1. Functionaliteiten: wat moest het prototype kunnen?
2. Omgeving: waar moest het prototype draaien aan het einde van de ontwikkelperiode?
3. Gebruikers en gebruiksvriendelijkheid: het aantal gebruikers en wat er minimaal nodig is voor deze gebruikers.
4. Experimenteren: welke mogelijkheden bieden de verschillende systemen, algoritmen, leermethoden en programmeertalen?

Door de keuzen langs vier assen te leggen werd niet alleen duidelijk waar je gezamenlijk aan werkt, maar ook wat je niet kon doen. Kies je voor meer functionaliteit, dan kun je minder aandacht besteden aan de gebruiksvriendelijkheid van het prototype. Leg je de nadruk op het experimenteren met verschillende machine learning pakketten, dan is de software voor langere tijd alleen beschikbaar op de laptop van de ontwikkelaar of in de lab-omgeving. Dit heeft weer invloed op het aantal gebruikers dat kan experimenteren.



Voorstel

Aan het einde van de sprint 6 (eind oktober)

- Functionaliteit: werkende software (die het Nationaal Archief zelf verder kan ontwikkelen op hun O/A-omgeving)
- Users: 10 users van het Nationaal Archief hebben toegang tot de software (gebruikers zullen 'begeleid' moeten worden)
- Omgeving: Nationaal Archief-medewerkers hebben toegang tot de ICTU Lab-omgeving om software tijdelijk te gebruiken
- Experimenteren: we hebben geëxperimenteerd met Python

Figuur 4 – de gemaakte keuzen aan het einde van sprint 6

2.3 Het experiment

Na de afronding van de ontwikkelfase en het opleveren van het prototype gingen we de experimenteerfase in. Tot deze fase rekenen we de installatie van het prototype op de testomgeving van het Nationaal Archief. Ook behoort het uitvoeren van het experiment met medewerkers van het Nationaal Archief ertoe. Het daadwerkelijk trainen van het classificatiemodel duurde twee maanden. De evaluatie van de resultaten en bevindingen bespreken we in een apart hoofdstuk, hoofdstuk 3.

2.3.1 Installatie prototype

Doel: het prototype installeren op de testomgeving van het Nationaal Archief.

In november 2017 is het prototype opgeleverd. Eind mei 2018 begonnen we met het trainen van het classificatiemodel. De trainingsperiode en daarmee het experiment liep tot 31 juli 2018. Dit was een periode van tegenslagen en overwinningen. Het prototype werkend krijgen op de infrastructuur van het Nationaal Archief was een uitdaging en duurde lang. We hebben meerdere malen op het punt gestaan het experiment te stoppen. We hebben continu de afweging gemaakt: volstaan de inspanningen, of worden deze te groot ten opzichte van wat het is: een experiment. Op cruciale momenten bereikten we telkens een doorbraak.

Dit zijn de voornaamste knelpunten die we tegenkwamen bij de installatie van het prototype:

1. **Integratie werkstroom van e-mailberichten met prototype**
   We hielden (te) lang vast aan het idee om het prototype te integreren in de werkstroom van e-mailberichten. Binnenkomende e-mailberichten van de deelnemers konden dan direct instromen in de database van het prototype. Dat zou een minimale belasting met een optimale opbrengst zijn. Deze koppeling was vooral beveiligingtechnisch onmogelijk. Gelukkig hadden we de 'Adidas koppeling[28]' ontwikkeld: een handmatige uploadmogelijkheid binnen het prototype. Omdat we ons in het begin van de installatie te veel focusten op het realiseren van de integratie, waren we uiteindelijk al een paar maanden verder voordat we tegen de volgende knelpunten opliepen.

2. **Technische installatie Docker container**
   De technologie die onder de lab-omgeving ligt, is Docker. Docker is een open source raamwerk, waarmee het mogelijk is om een applicatie in een lichtgewicht, verplaatsbare container te verpakken. Zo wordt een applicatie op een server installeren even eenvoudig als een mobiele app installeren op je tablet of smartphone[29]. Het prototype was verpakt in zo'n container en opgeleverd aan het Nationaal Archief, klaar voor installatie. Simpel, zou je zeggen. Het Nationaal Archief zorgt ervoor dat er een Docker-omgeving aanwezig is en het prototype installeer je met een eenvoudige klik op de knop.

   Het was allesbehalve een eenvoudige installatie. De container was namelijk niet goed opgebouwd. Dit betekende een reconstructie van de code om de informatie die in de container was opgeslagen, weer leesbaar te krijgen. Na de reconstructie kwamen we erachter dat we enkele cruciale onderdelen van het prototype misten, waaronder het getrainde classificatiemodel. Het missende classificatiemodel was overigens te verantwoorden. De container met code stond op een openbare plek. Als het getrainde model daar in had gezeten, was het mogelijk geweest om de e-mailberichten van de training te reconstrueren vanuit de database. Dit zou een inbreuk zijn op de gegevensverwerkingsovereenkomst.

   Door eerst tijd te stoppen in de automatische verwerking van de informatiestroom – een focus op de eigen infrastructuur en wat we daar kunnen - hebben we het opgeleverde product niet gecontroleerd op volledigheid en werking. Tegen de tijd dat we erachter kwamen dat de Docker container niet goed was geconstrueerd en er onderdelen misten, was het ingehuurde ontwikkelteam niet meer beschikbaar. De oplossing was een eigen reconstructie maken met een kopie van de demoversie die nog aanwezig was in de lab-omgeving. Dit heeft relatief veel tijd gekost. Er is aandacht nodig voor checks op werking en volledigheid, zelfs als het gaat om een prototype of een experiment.

3. Prioriteit experimenteren in de organisatie

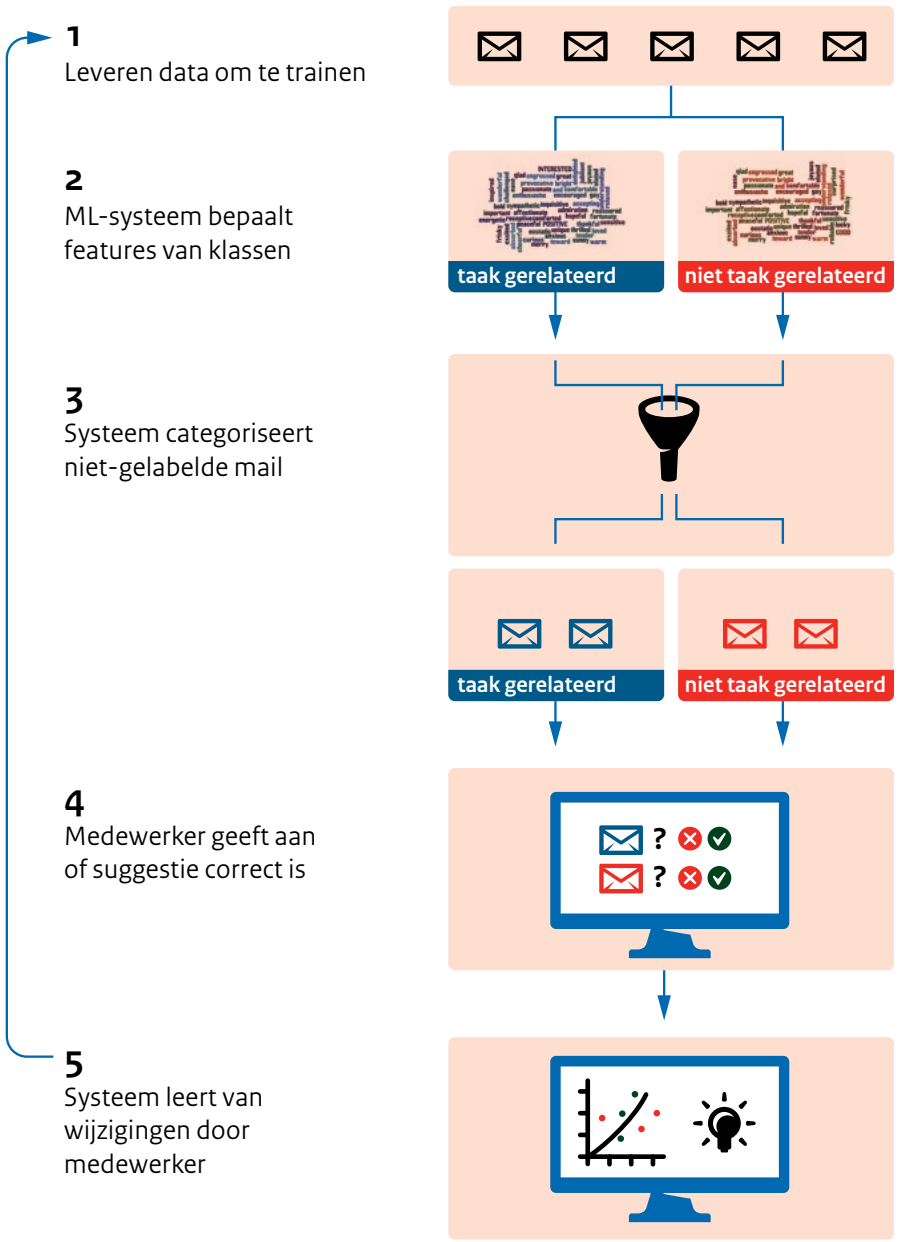
Los van de technische problemen was de tijd die we aan de installatie konden besteden beperkt. Dit was verklaarbaar door prioritering van dagelijkse werkzaamheden. Deze werkzaamheden gaan voor op een experiment. Juist in de periode van de ontwikkeling van het prototype en het experiment lagen er voor de afdeling Technisch Beheer enkele grote klussen (zoals opleveren van de nieuwe websites).

Wil je experimenteren binnen je organisatie? Dan is het belangrijk dat je dit zoveel mogelijk met eigen medewerkers doet. Zo blijft kennis in huis. We probeerden zo vroeg mogelijk in het proces de afdeling Technisch Beheer te betrekken, maar het bleef lastig. Een optie om dit beter te laten verlopen, is nadenken over een vast experimenteermoment of een vaste structuur. Daarbij is experimenteren onderdeel van je werkzaamheden. Hierdoor komt de druk minder op dat ene moment te liggen. Dit voorkomt (onderlinge) frustraties.

2.3.2 Het experiment

Doel: verder trainen van het classificatiemodel en het scheppen van vertrouwen en transparantie in zelflerende systemen.

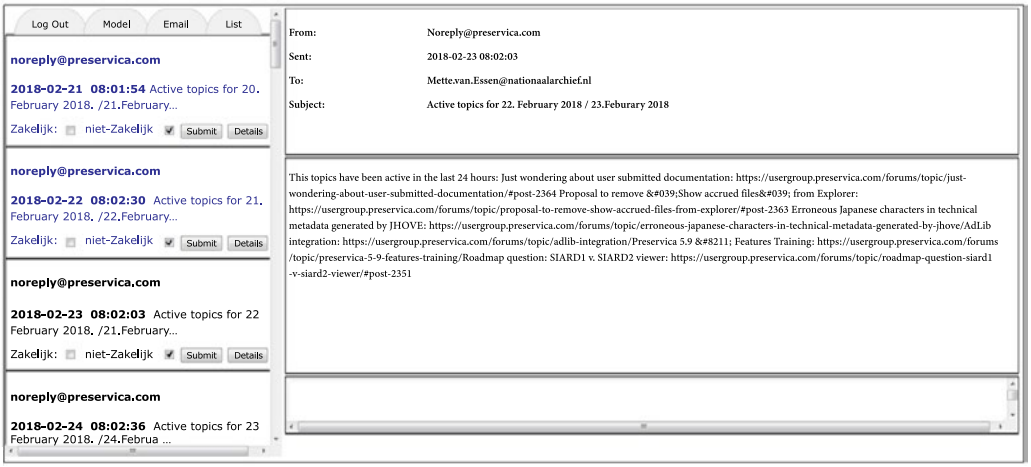
Het prototype werkt als volgt. De medewerker voedt het systeem met eigen (van tevoren geselecteerde) e-mailberichten. Het systeem maakt een voorspelling op basis van wat het in de ontwikkelfase heeft geleerd. Via de web-interface[30] krijgt de medewerker een voorspelling van de klasse te zien (ruis of functioneel). Vervolgens geeft de medewerker aan of deze voorspelling correct is. De medewerker geeft dit terug aan het systeem. Het systeem leert vervolgens van de wijzigingen die de medewerker doorgeeft[31]. Dit iteratieve proces herhaalt zich zolang er nieuwe e-mailberichten worden ingelezen.



Figuur 5 - werking prototype

Het classificatiemodel bestond uit drie verschillende algoritmen: het Multinominal Naive Bayes (MNB) algoritme (de zogenaamde bag of words), het Random Forest (RF) algoritme en het Extreme Random Forest (ERF) algoritme (beslisbomen). Deze algoritmen trainden we afzonderlijk. Het trainen gebeurde met voorspellingen die de deelnemers controleerden. We besloten om telkens het slechtst functionerende algoritme te trainen.

Over de gehele periode trainden we het classificatiemodel zeventien keer. We startten met trainen vanaf het moment dat het prototype op de NA-omgeving was geïnstalleerd. De eerste trainingen deden we met de controleset, die aanwezig was vanuit de ontwikkelperiode. Vervolgens hebben we nieuwe, ongelabelde e-mailberichten in het prototype ingelezen. Hiermee controleerden we de werking van het prototype. Ook werkten we de werkwijze voor het experiment verder uit. In de periode van 17 mei tot 5 juni startten alle deelnemers (vijftien in totaal) met het experiment. Het experiment liep officieel tot en met 31 juli. Helaas crashte de installatie van het prototype op 30 juli. We hebben gekeken naar een fix, maar de tijdsinvestering voor de ene dag was te groot. Hierdoor hebben we de eindstreep net niet gehaald. Het was de bedoeling om alle algoritmen nog een keer te trainen na afloop, maar dit is niet gelukt.



Figuur 6 - scherm met voorspelling e-mailberichten

2.3.3 Evaluatiemogelijkheden binnen prototype

Doel: inzicht geven in het functioneren van algoritmen en het opbouwen van transparantie en vertrouwen.

In de ontwikkelperiode hebben we veel nagedacht en gebrainstormd over de visualisatiemogelijkheden. Het belangrijkste idee achter de visualisaties en schermen was om inzicht te verkrijgen in hoe het classificatiemodel en de afzonderlijke algoritmen de e-mailberichten beoordeelden. Daarnaast moest het mogelijk zijn om zonder tussenkomst van een expert het functioneren van de afzonderlijke algoritmen te beoordelen.

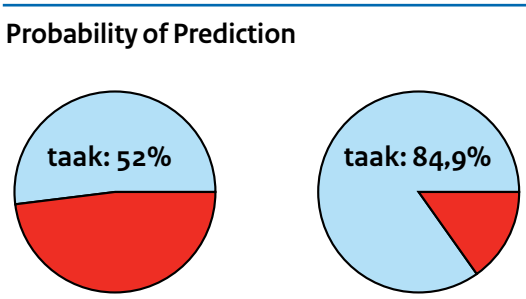
Naast de simpele weergave van de voorspelling in het basisscherm (zie fig.6) was er de mogelijkheid de voorspelling in detail te bekijken. In dit aparte scherm was een visualisatie van de beoordeling per algoritme te zien op verschillende manieren.

Voorbeeld Ruis e-mailbericht

Voorbeeld Functioneel e-mailbericht

Figuur 7 - e-mailberichten in detailscherm

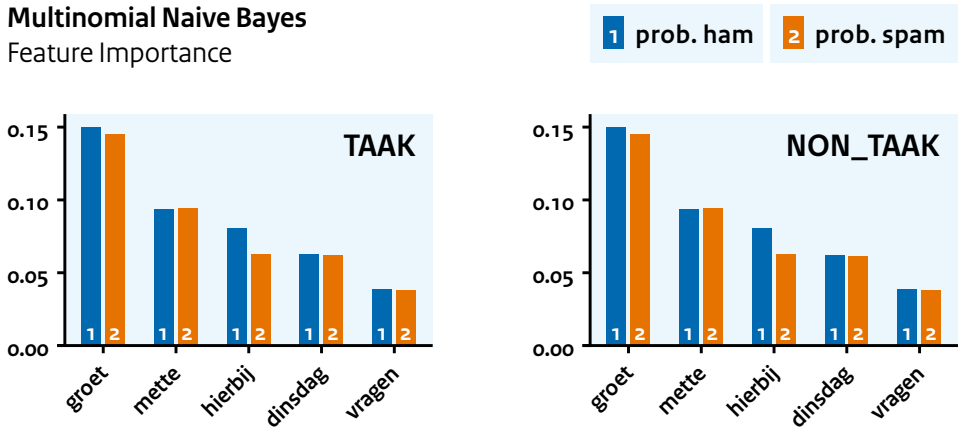
Gekleurde woorden en woordcombinaties (zie fig.7) geven de belangrijkste woorden aan waarop de algoritmen hebben besloten dat het e-mailbericht bij een bepaalde klasse hoort. Hoe groter en dikker de woorden, des te belangrijker ze zijn. De ongemarkeerde woorden zijn óf niet gebruikt óf hebben een verwaarloosbare waarde.



Figuur 8 - cirkeldiagram 'Probability of Prediction'

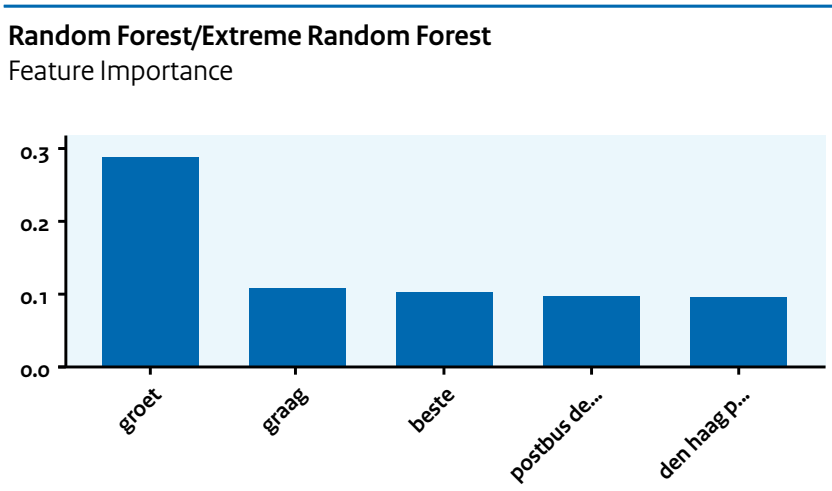
Het cirkeldiagram 'Probability of Prediction' (fig.8) geeft weer met welke zekerheid het algoritme een e-mailbericht aan een bepaalde klasse toewijst. Is dit percentage ergens in de 50%, dan is het een goede gok. Met 80 tot 90% is de zekerheid van het algoritme vrij hoog. Het idee achter deze visualisatie was dat je de medewerker eerst de e-mailberichten voorschotelt waarover het algoritme twijfelt (50-60%). Als het vertrouwen genoeg is opgebouwd, kun je een actie verbinden aan de e-mailberichten met een zekerheids-voorspelling van boven de 80%. De aanname is dat het "grijze gebied" van 50-60% steeds kleiner wordt en het vertrouwen in de voorspellingen steeds groter. Na verloop van tijd kan het algoritme steeds meer classificaties automatisch verwerken.

De laatste visualisatie in dit scherm is het diagram Feature Importance. Dit zijn de vijf belangrijkste kenmerken/woorden die een rol speelden bij de classificatie van het specifieke e-mailbericht (dus niet van de gehele gegevensset).



Figuur 9a - diagram van het Multinominal Naive Bayes algoritme

Bij het diagram van het Multinominal Naive Bayes algoritme (fig.9a) zie je balken met een waarde. Deze waarden vertegenwoordigen de kans dat de bijbehorende woorden in elke klasse voorkomen. Dit maakt een vergelijking mogelijk. Bovenstaande afbeelding is van het begin van het experiment. Door het lage aantal gegevens zijn de verschillen tussen de vijf belangrijkste woorden (nog) nauwelijks te onderscheiden. Hieruit kun je concluderen dat de beslissing die het algoritme maakt, nog is bepaald door een groot aantal woorden. In de loop van het experiment werd het onderscheid tussen de twee balken groter.

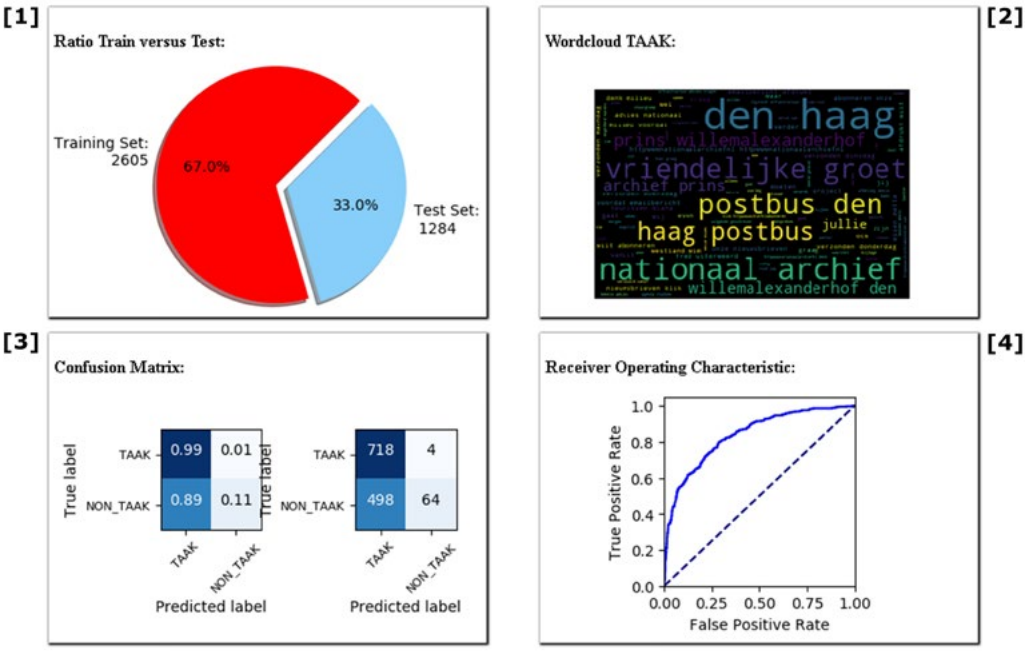


Figuur 9b - diagram van het Multinominal Naive Bayes algoritme

Voor de Forest algoritmen (fig.9b) ziet het diagram er iets anders uit. Deze algoritmen maken gebruik van een beslisboom. De waarde van deze balken is het aantal keer dat elk woord in het beslissingsproces wordt opgenomen. Dat wil zeggen dat het woord een splitsing (node) in de beslisboom bepaalt.

Het modelscherm

Het andere scherm, het modelscherm of het dashboard, ontwikkelden we om de feitelijke prestaties van de algoritmen te evalueren. Het scherm heeft vier visualisatie- en evaluatiemogelijkheden:



Figuur 10 - modelscherm prototype

1. **Ratio Train versus Test**: dit figuur geeft per algoritme de hoeveelheid e-mails weer die we gebruikten voor de training van het algoritme. De taartdiagram is onderverdeeld in een Training Set (de e-mail-berichten waarvan de klasse bevestigd is en die we gebruiken om het model te trainen) en de Test Set (de e-mailberichten waarvan de klasse bevestigd is en die we gebruiken om de werking van het classificatie-model te controleren).

2. **Wordcloud TAAK:** de Wordcloud representeert de frequentie van de woorden die voorkomen in de e-mailberichten die tot de klasse TAAK (of functioneel) behoren. Hoe vaker het woord of de combinatie van woorden voorkomt, hoe groter het woord is weergegeven. Stopwoorden zijn eruit gefilterd.

3. **Confusion Matrix**: de confusion matrix is een tabel die de prestaties van een classificatiemodel beschrijft. Dit gebeurt op basis van een reeks gegevens waarvan de feitelijke waarde bekend is: de controleset. De confusion matrix geeft weer in hoeverre het classificatiemodel verward of 'confused' is.

4. **Receiving Operating Curve** (**ROC**): is een andere manier om de nauwkeurigheid van het algoritme te meten. Hoe dichter de curve bij de gestippelde diagonale lijn ligt, des te meer gerandomiseerd het model is. Dat wil zeggen dat het een slechte nauwkeurigheid heeft.

2.4 Samenvattend

Buiten de resultaten, die we beschrijven in het volgende hoofdstuk, heeft het experiment ons veel geleerd. Niet alleen over de technologie, maar voornamelijk over wat het inhoudt om een experiment met een nieuwe technologie als deze uit te voeren.

De belangrijkste lessen:

- **Maak privacy onderdeel van je experiment**
  Je moet nadenken over gegevensverwerking, vooral als een systeem dit gaat doen. Het organiseren van gesprekken, het opstellen van maatregelen en het bij elkaar brengen van de juiste personen kost tijd. De verwerking van (persoons)gegevens mag geen excuus zijn om een experiment niet uit te voeren.

- **Zelf ontwikkelen van een prototype helpt bij concretiseren van een probleem**
  Met dit experiment hebben we niet alleen geleerd hoe zelflerende systemen werken. We hebben ook beter inzicht gekregen in de problematiek die speelt rondom e-mail en ongestructureerde informatie in het algemeen.

- **De huidige infrastructuur, privacymaatregelen en de inrichting van organisatieprocessen brengen restricties voor het experimenteren mee**
  Gebruik deze restricties als randvoorwaarden voor het uitvoeren van je experiment. Probeer tijdens het uitvoeren na te denken hoe het anders kan. Hoe zou je hetzelfde doen in een ideale situatie? Plot dit op de echte situatie en kijk wat er in de toekomst haalbaar is.

- **Timeboxen en visualisaties helpen bij het maken van de juiste keuze**
  De beperkte ontwikkelperiode en de visualisatie van te maken keuzes heeft ons geholpen in een korte tijd een werkend prototype op te leveren. Het heeft bijgedragen aan een geslaagd eindresultaat en inzichtelijk gemaakt wat wel en niet kan.

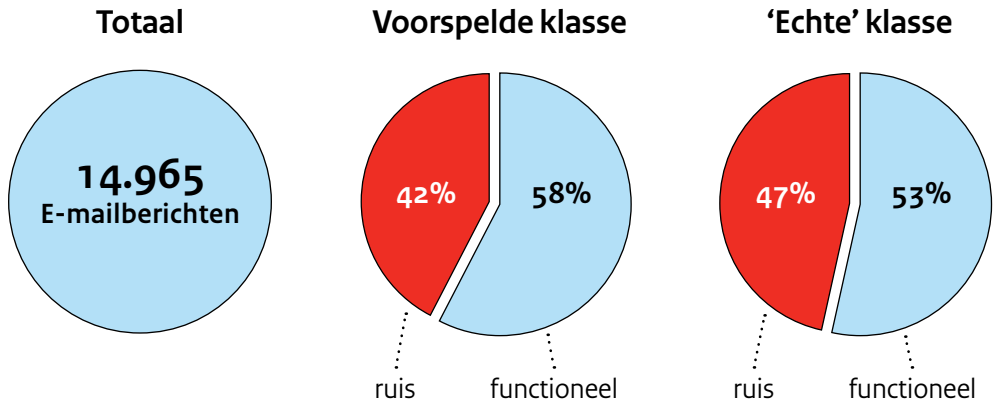
3. Resultaten experiment

In het vorige hoofdstuk hebben we de evaluatiemogelijkheden van het prototype besproken. Gedurende het gehele experiment hield de projectleider met deze schermen het functioneren van de algoritmen en het classificatiemodel in zijn algemeenheid bij. Een deel van de analyses deden we met de e-mailberichten van de projectleider zelf.

In dit hoofdstuk gaan we gedeeltelijk in op de technische analyse. Het gaat dan voornamelijk over de resultaten die horen bij de doelstellingen van het experiment (zie pag.15). Aan de hand van het prototype hebben we zelf ervaren hoe het is om met een zelflerend systeem te werken. Wat is er mogelijk, wat werkt goed en wat werkt er niet? We ontwikkelden kennis en hebben een beter zicht op wat er nodig is in de organisatie om een systeem als dit te laten draaien.

3.1 Feitelijke analyse

In totaal hebben het classificatiemodel en de medewerkers 14.965 e-mailberichten respectievelijk geclassificeerd en gecontroleerd. Het classificatiemodel voorspelde dat 8.613 e-mailberichten (58%) tot de klasse functionele e-mail behoorden en 6.352 e-mailberichten (42%) tot de klasse ruis e-mail. Uiteindelijk labelden de medewerkers 7.970 e-mailberichten (53%) als functionele e-mail en 6.995 (47%) e-mailberichten als ruis e-mail.



Figuur 11 - verdeling klassen in de gegevensverzameling

Dit zegt iets over de verdeling van de twee klassen in de e-mailverzameling, maar niets over de nauwkeurigheid van het classificatiemodel. Dit konden we niet voor het classificatiemodel aflezen, maar wel voor de algoritmen afzonderlijk.

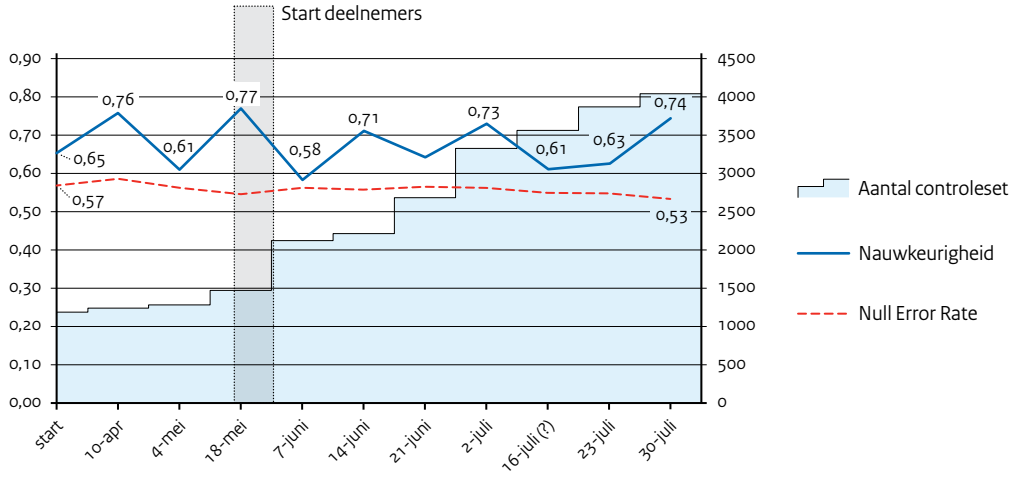
3.1.1 Functioneren van algoritmen

Het classificatiemodel is binair. Dat wil zeggen dat er twee antwoorden mogelijk zijn: ja of nee. Ruis e-mailberichten hebben de positieve waarde gekregen. Het uitgangspunt was dat we de ruis uit de verzameling e-mailberichten wilden filteren, zoals een spamfilter spam uit je e-mail filtert. De vraag die we aan het classificatiemodel stelden, was: 'Is het e-mailbericht ruis, ja of nee?" [32].

We bespreken het functioneren van de drie algoritmen kort aan de hand van de volgende waarden, namelijk
1. **Nauwkeurigheid** (Accuracy): Hoe vaak heeft het algoritme de voorspelling goed?
2. **True Positive Rate** (Sensitivity): Hoeveel (het percentage) ruis e-mailberichten zijn correct gelabeld?
3. **True Negative Rate** (Specificity): Hoeveel (het percentage) functionele e-mailberichten zijn correct gelabeld?

Multinominal Naive Bayes (MNB)
Het MNB algoritme had de meeste schommeling in waarden en is daarom het meest getraind[33]. Als je kijkt naar de begin- en eindwaarde, is de nauwkeurigheid verbeterd van 66% naar 74%. Als je kijkt over de gehele periode, dan zijn er veel pieken en dalen in de nauwkeurigheid.

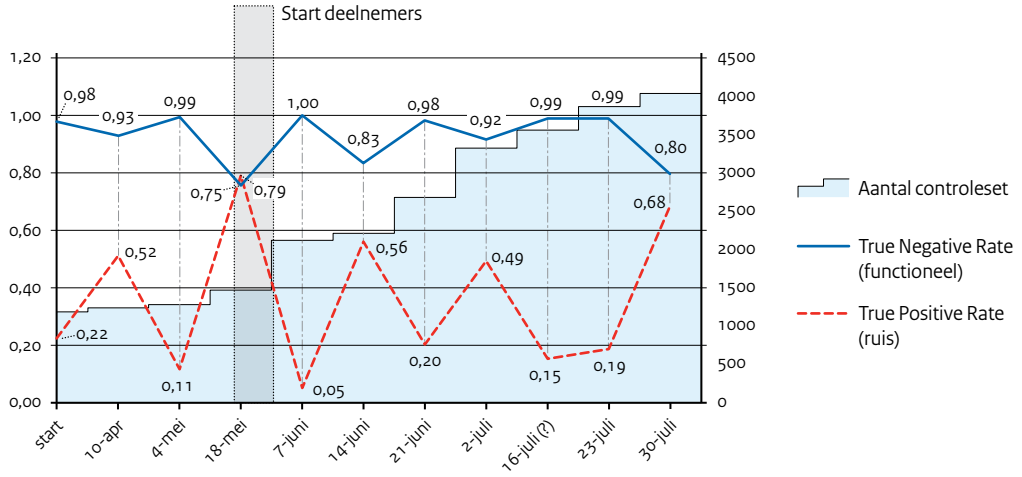
Multinominal Naive Bayes - Nauwkeurigheid



Figuur 12 - Nauwkeurigheid Multinominal Naive Bayes algoritme

De rode lijn in de grafiek geeft de null error rate weer. Dit wil zeggen: hoe vaak heeft het algoritme het mis als elk e-mailbericht een positieve waarde (label ruis) krijgt? Op verschillende momenten ligt de nauwkeurigheid maar net boven deze baseline. De dalingen op 4 mei en op 7 juni[34] zijn te verklaren. Dit waren namelijk momenten waarop we nieuwe e-mailberichten van medewerkers toevoegden. Door een nieuwe manier van beoordelen van de klassen (door de medewerker) en eventueel ander taalgebruik in de e-mailberichten, had het algoritme tijd nodig om zich te stabiliseren. Het was duidelijk aan het leren. Hetzelfde gebeurt als je kijkt hoe goed het algoritme functionele en ruis e-mailberichten voorspelt. Het algoritme heeft de meeste moeite met ruis e-mailberichten correct voorspellen. Dit heeft een logische verklaring. Ook de deelnemers aan het experiment hebben meer moeite om ruis e-mailberichten te classificeren (zie hfst.3.2.2 Juistheid van de voorspelling). Het is voor een medewerker makkelijker in te schatten wat een functioneel e-mailbericht is. Ruis is daarmee moeilijker te bepalen.

Multinominal Naive Bayes



Figuur 13 - Percentage correct gelabelde e-mail (Functioneel/Ruis)

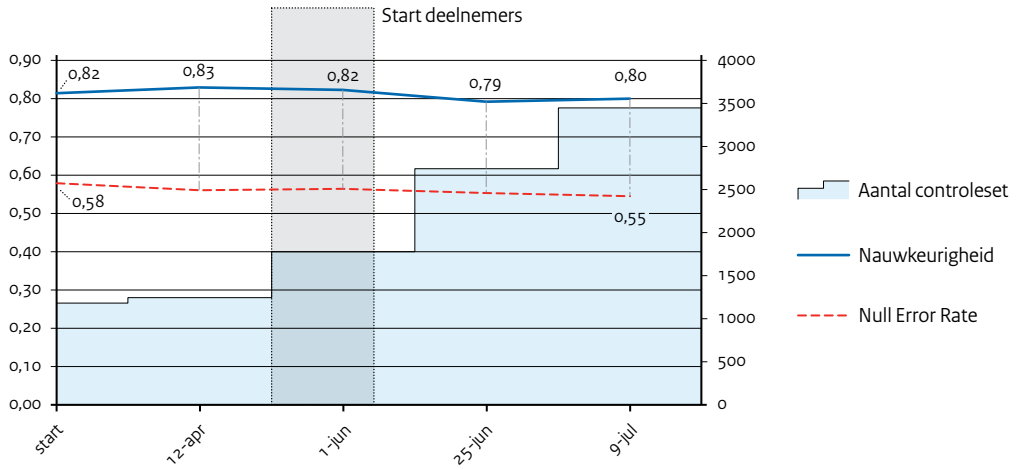
Na elke training bekeken we nieuwe e-mailberichten in detail[35]. We keken naar de belangrijkste features (zie hfst. 3.1.2. Features en andere kenmerken) en de zekerheid van de voorspelling (zie hfst.2.3.3 Evaluatie mogelijkheden binnen prototype). Uit de steekproeven bleek dat de meeste voorspellingen (van beide klassen) van het MNB algoritme bleven hangen op een zekerheid tussen de 50% en de 59%. Dat is niet veel beter dan een willekeurige gok. In de loop van het experiment werd dit iets hoger en lagen de percentages boven de 55%.

	40-49.9%	50-59.9 %	60-69.9%	70-79.9%	80-89.9%	90-99.9%	100%
MNB	**16.5%**	**80.7%**	**2.8%**				

Forest algoritmen

De resultaten van de andere twee algoritmen, het Random Forest (RF) algoritme (vier keer getraind) en het Extreme Random Forest (ERF) algoritme (drie keer getraind) liggen dicht bij elkaar. De waarden zijn stabieler dan die van het MNB algoritme. De nauwkeurigheid van deze algoritmen lag voor de gehele periode rond de 80%. Er zijn geen grote schommelingen zichtbaar, ook niet in de periode dat de deelnemers startten met het experiment.

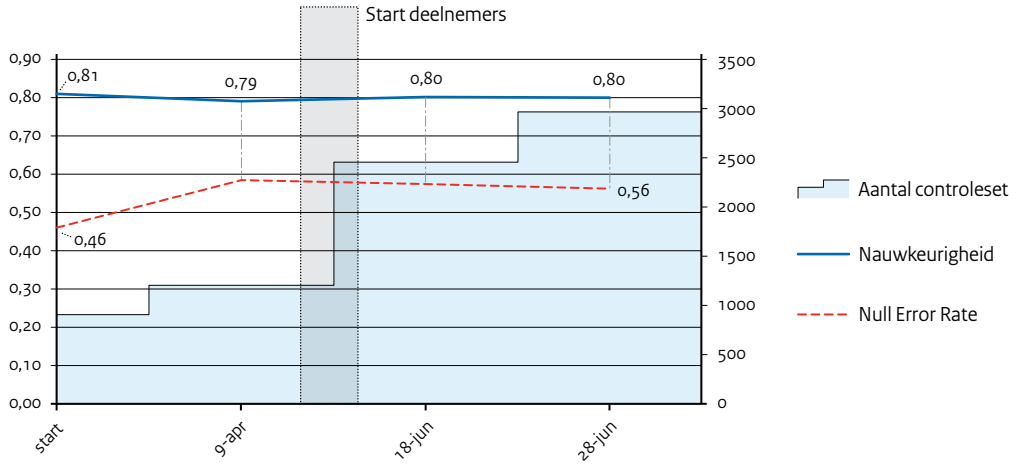
Random Forest - Nauwkeurigheid



Figuur 14 - Nauwkeurigheid Random Forest algoritme

Bij het ERF algoritme valt op dat de null error rate bij de start van het experiment lager ligt dan bij de andere algoritmen. Dit wil zeggen dat de initiële controle set een groter aantal ruis e-mailberichten (54%) bevatte dan in de loop van het experiment, waar het gemiddelde rond de 44% lag. Waarom dit alleen bij het ERF algoritme is, is onverklaarbaar.

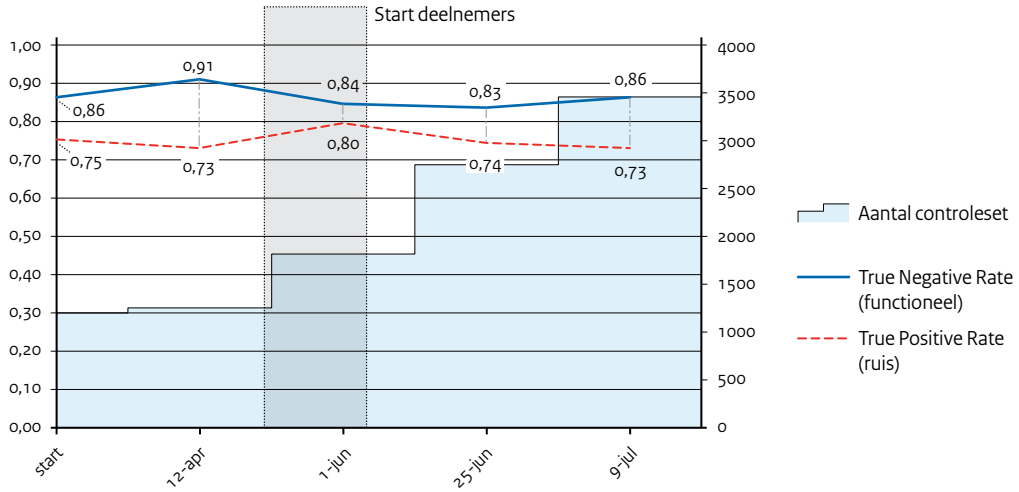
Extreme Random Forest - Nauwkeurigheid



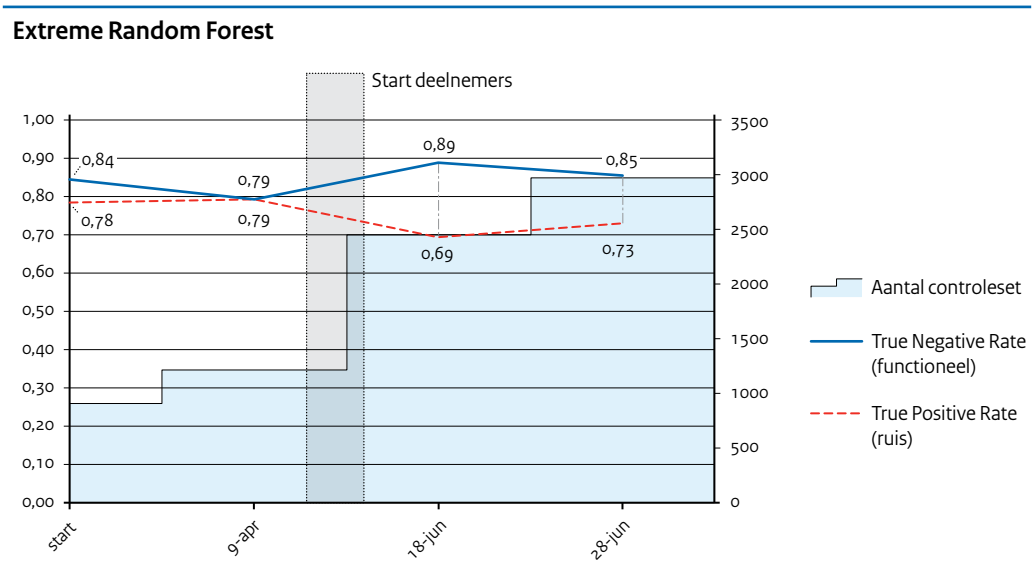
Figuur 15 - Nauwkeurigheid Extreme Random Forest algoritme

De correctheid van de classificatie van zowel de functionele e-mailberichten als de ruis e-mail berichten ligt bij deze algoritmen veel dichter bij elkaar. Ook hier is minder schommeling te zien. Net als bij het MBN algoritme zijn de Forest algoritmen beter in de classificatie van functionele e-mailberichten. Er zijn lichte dalingen en stijgingen te zien op de momenten dat we onbekende e-mailberichten hebben ingelezen.

Random Forest



Figuur 16 - Percentage correct gelabelde e-mail (Functioneel/Ruis)



Figuur 17 - Percentage correct gelabelde e-mail (Functioneel/Ruis)

Omdat de nauwkeurigheid van beide algoritmen ongeveer hetzelfde bleef, rond de 80%, is het de vraag of deze algoritmen beter zijn geworden door training. Waar aan het begin van de trainingsperiode de waarschijnlijkheid (zekerheid) van de voorspellingen nog veel rond de 50-60% lag, was de zekerheid van de voorspellingen aan het einde van de periode een stuk hoger. Dit wijst erop dat het trainen effect had. Gemiddeld liggen de percentages bij het RF algoritme voornamelijk tussen de 60 en 69,9% en bij het EFR algoritme tussen de 70 en 70.9%. Beide hebben aan het einde van de periode uitschieters naar een waarschijnlijkheid tussen de 80 en 100%. Nieuwsberichten scoorden hoog in percentage voor de ruis e-mailberichten en duidelijk functionele e-mailberichten scoorden hoog bij de functionele klasse.

	40-49.9%	50-59.9 %	60-69.9%	70-79.9%	80-89.9%	90-99.9%	100%
RF	1.4%	22.1%	25.5%	17.9%	14.5%	12.4%	6.2%
ERF		15.9%	17.2%	26.2%	17.9%	15.2%	7.6%

3.1.2 Features en andere kenmerken

Het classificatiemodel deed voorspellingen aan de hand van de tekst van het e-mailbericht (de body). Hierbij hebben we ons verkeken op de e-mailhandtekening die prominent aanwezig was in de meeste e-mailberichten. Deze stelt een medewerker vaak automatisch in. Het is mogelijk om de woorden van de e-mailhandtekening eruit te filteren. Uiteindelijk hebben we besloten om dit niet tijdens het experiment te doen, maar af te wachten of er een verandering in de woorden zou optreden. Dit gebeurde niet. Omdat het prototype de eindstreep niet heeft gehaald, konden we dit ook niet achteraf doen.

Figuur 18 - De wordclouds van de MNB, RF en ERF algoritmen

De e-mailhandtekening had een effect op de voorspellingen, vooral bij het MNB algoritme. Dit algoritme voorspelde mail met een e-mailhandtekening vaker als functionele e-mail, wat niet altijd het geval hoeft te zijn. Hoofdzakelijk gebeurde dit bij e-mailberichten waar onvoldoende onderscheidende woorden aanwezig waren. De forest algoritmen hadden hetzelfde probleem op een andere manier, namelijk met de woorden 'verstuurd vanaf iphone'. Deze e-mailberichten beoordeelden de beslisbomen vaker als ruis.

Wat opvalt uit de wordcloud, is dat wij (medewerkers van het Nationaal Archief) in onze functionele e-mailberichten veel woorden als 'jullie', 'wij' en 'jij' gebruiken. We e-mailen heel amicaal. Voor de functionele mail zag je in de loop van het experiment dat woorden als 'stukken', 'hierbij' en 'versie' zich steeds meer onderscheidden als sterke feature voor functionele e-mail.

Een opvallend sterke feature voor ruis e-mail werd 'Beste'. Dat is misschien niet gelijk een voor de hand liggend kenmerk van dit type e-mailberichten. De verklaring is als volgt. Uitnodigingen en algemene berichten beginnen we vaak met Beste collega's of Beste relatie. Daar tegenover staat dat 'Hoi' en 'Hallo' sterke features zijn voor functionele e-mailberichten. Dat is waarschijnlijk omdat een e-mailbericht gericht aan een specifiek persoon vaker functioneel is. Je vraagt die persoon iets te doen of een bepaalde actie uit te voeren: 'Hoi …, Hierbij de stukken. Kun je deze…'

Figuur 19 - voorbeeld van 'Beste' als feature

3.2 Vertrouwen en transparantie

We spreken veel over vertrouwen en transparantie in de context van zelflerende systemen en algoritmen. De betekenis ervan is vaak onduidelijk. Met dit experiment probeerden we dit voor onszelf vorm te geven. Wat is belangrijk bij het vaststellen van vertrouwen? Heeft het effect als een medewerker zelf het model traint? Moeten de beslissingen die een algoritme maakt op een bepaalde manier worden weergegeven of inzichtelijk gemaakt? En wat is er nodig om een zelflerend systeem daadwerkelijk binnen een organisatie te gaan gebruiken? We hebben de deelnemers van het experiment bevraagd over deze onderdelen.

3.2.1 Algemene indruk

De deelname aan het experiment ervaarde iedereen als positief. Kennismaken met een nieuwe technologie en dit zelf toepassen geeft meer inzicht in een nieuw mechanisme. Het wordt meer dan theorie alleen. Je hebt niet het gevoel dat je bij nieuwe ontwikkelingen vanaf de zijlijn meekijkt. De eenvoud van het trainen en de relatief kleine inspanning maakten dat het goed te combineren was met de dagelijkse werkzaamheden. Sommige medewerkers ervaarden het trainen na verloop van tijd wel als saai, omdat het ging om het uitvoeren van een repetitieve taak.

Deelname aan het experiment was op vrijwillige basis. Het was niet moeilijk om genoeg deelnemers te vinden. Wat motiveerde hen? Dat was vooral belangstelling naar de werking van zelflerende systemen en

nieuwsgierigheid naar of de uitkomsten in de praktijk bruikbaar zijn. Medewerkers van het Nationaal Archief beseffen het belang van de inzet van technologie binnen informatieprocessen. Het experiment, en daarmee de eigen deelname, zagen ze als een eerste logische stap. We moeten technologische mogelijkheden veel beter benutten, zo was de opvatting. Dat moet niet alleen bij het Nationaal Archief zelf, maar ook binnen het gehele archief- en informatienetwerk. Dit is de reden dat sommige deelnemers benieuwd zijn hoe een gelijkwaardig experiment met grotere aantallen (en eventueel andere organisaties) zou uitpakken. Krijg je dan een beter en/of stabieler model en wat is de problematiek waar je tegenaan loopt bij een schaalvergroting?

3.2.2 Juistheid van de voorspelling

Juistheid bij aanvang van trainingsperiode
De eerste indruk van de juistheid van de voorspellingen verschilde per persoon. Het varieerde van "zat er nogal eens naast" en "viel me eigenlijk tegen" tot "niet slecht voor een machine" en "over het algemeen verrassend goed". De grootste groep, tweederde van de deelnemers, was positief gestemd en had een goede eerste indruk. Niet bij elke mail was een correctie nodig, wat bij sommige deelnemers wel de verwachting was. "Ik was zo onder de indruk van de mate van juistheid dat ik de neiging had om de computer/het systeem gelijk te geven."

Eenderde van de deelnemers keek er anders tegenaan. Ook hier speelden persoonlijke verwachtingen een belangrijke rol. Er was bijvoorbeeld de verwachting dat het systeem aan het begin van het experiment initieel beter zou zijn. Enkele deelnemers hadden de ervaring dat een foutieve voorspelling gekoppeld werd aan een bepaalde afzender (intern of extern). Omdat het classificatiemodel enkel keek naar de inhoud van de e-mailberichten en niet naar afzenders, kan dit duiden op ander taalgebruik bij deze specifieke afzenders.

Buiten fouten specifiek op één afzender, viel het de deelnemers op dat veel foutieve voorspellingen voorkwamen bij gemixte e-mailberichten. Voorbeelden hiervan zijn een functionele e-mail die ook persoonlijke communicatie tussen collega's bevat en de zogenaamde twijfelgevallen (is het functioneel of is het ruis?). Foute voorspellingen bij dit soort e-mailberichten namen de deelnemers het systeem niet kwalijk. "In het begin toch nog wel wat ruis. Daarbij wel…als het fout was, was het wel logisch dat de fout gemaakt werd. Dit waren namelijk ook effectief twijfelgevallen voor mijzelf." en "Het is niet vreemd dat als de mens de machine/het systeem traint deze in hetzelfde 'grijze gebied' als de mens fouten maakt".

Juistheid aan einde van trainingsperiode
Aan het einde van de trainingsperiode merkte ongeveer de helft van de deelnemers verbetering in de juistheid van de beoordeling. In de loop van het experiment ervaarden ze het systeem als 'slimmer'. Of dit een feitelijke verbetering was is door de meeste deelnemers niet geverifieerd (zie hfst 3.2.3 het trainen van het model). De verbetering was gevoelsmatig: "Ik had wel de indruk dat er een verbetering in zat. Weet niet hoe dit procentueel zat. Kan natuurlijk ook zijn dat mijn eigen patroon of eigen consequent beoordelen verbeterde in de loop van het experiment." Een foutieve voorspelling kwam in beide klassen voor. Niemand heeft specifiek het idee gehad dat in één van de twee klassen meer fouten voorkwamen. Het viel op dat het classificatiemodel in de loop van het experiment bepaalde 'soorten' e-mailberichten steeds beter herkende (zoals nieuwsbrieven, puur zakelijk enz.).

Trainen van het classificatiemodel heeft duidelijk invloed. De beleving van de juistheid van de voorspelling heeft voor een groot gedeelte te maken met het persoonlijk verwachtingspatroon van de deelnemer, en minder met de feitelijke juistheid.

3.2.3 Het trainen van het model

We vroegen aan de deelnemers hoe zij het prototype gebruikten tijdens het experiment, met name de schermen[36]. Bijna geen enkele deelnemer heeft het detail en/of modelscherm (zie hfst 2.3.3 Evaluatie-mogelijkheden binnen het prototype) vaak bekeken. De meeste deelnemers keken niet naar deze schermen. Buiten het niet gebruiken misten ze de informatie niet. Soms keek een deelnemer uit nieuwsgierigheid. Een enkele keer was dat omdat hij of zij niet zeker was tot welke klasse het e-mailbericht behoorde. Als een deelnemer keek, was het een extra check. Het was dus een aanbeveling van het systeem, niet een manier om de werking van het systeem te controleren.

Buiten de praktische reden – details zien vergde een extra handeling – betwijfelden de deelnemers of ze (meer) hadden gekeken als het makkelijker was. Details waren niet nodig omdat:
1. Het ging om recente e-mailberichten. Deze zitten vers in het geheugen. In één oogopslag weet je of de classificatie goed of fout is.
2. De verantwoordelijkheid voor het trainen ligt bij jezelf.
3. Details kunnen je afleiden en/of je keuze beïnvloeden. Wat zie ik? Begrijp ik dit? Wil ik dit begrijpen?

Deelnemers gaven aan dat een samenvatting van de voorspellingen achteraf toegevoegde waarde kan hebben. Je geeft op die manier een update over het functioneren van de algoritmen. Dit hoeft niet op detailniveau. Het kan bijvoorbeeld over een bepaalde periode gaan. Voorwaarde hiervoor is dat het op een begrijpelijke manier gebeurt. De verantwoordelijke persoon voor het systeem geeft een toelichting of denkt na over meer gebruiksvriendelijke visualisaties. Een interactie met de visualisatie en/of de rapportage beval een enkeling aan. Bij een wordcloud wil je bijvoorbeeld woorden kunnen uitfilteren en/of inzoomen op één woord en misschien de onderlinge relaties zien.

3.2.4 Vertrouwen in het systeem

Ongeveer de helft van de deelnemers had vertrouwen in het systeem. De andere helft was iets sceptischer. Geen één medewerker had absoluut nul vertrouwen.

De groep deelnemers die minder vertrouwen had in het systeem, vond dat er nog het één en ander moest gebeuren om het vertrouwen te vergroten. Zonder de mogelijkheid van een interventie lieten de twijfelaars het systeem nog niet zijn gang gaan. De voorspellingen hadden in deze fase nog te weinig nuance. Het systeem maakte soms belangrijke fouten. Een voorbeeld hiervan was mail die een deelnemer wilde bewaren, maar die het model als ruis classificeerde[37]. Dit is niet te voorkomen en je kunt nooit met 100% zekerheid zeggen dat dit niet gaat gebeuren. Je kunt hier wel op trainen. Al roept dit ook vragen op. Voor deze simpele vraag werkte het, maar was de vraag niet te simpel? Heeft functioneel belangrijk en functioneel niet-belangrijk geen grotere waarde? Accepteren we dit als het gaat tussen het onderscheid belangrijk/onbelangrijk? Het zoeken naar het juiste antwoord op de juiste vraag blijkt cruciaal bij een zelflerend systeem.

De deelnemers die wel (genoeg) vertrouwen hadden in het systeem, hadden dit om verschillende redenen. Het systeem was niet slechter of beter dan de mens. Daarnaast is het allang bewezen dat systemen als deze beslissingen kunnen maken. Voor sommige beslissingen kan een machine dit waarschijnlijk beter, of in ieder geval consequenter, inschatten dan wijzelf. Het zal best weleens misgaan, maar dat gebeurt nu ook. De mens doet het helemaal niet zo goed. Een systeem dat consequent beoordeelt, maar af en toe fouten maakt, is daarom misschien wel beter. Medewerkers zijn tot dit inzicht gekomen, omdat ze tijdens de trainingsperiode zijn geconfronteerd met de eigen inconsequentie. Hoe beoordeelde ik dit bericht gisteren? Hoe leer ik een machine iets als ik het zelf al niet goed weet?

Zelf trainen helpt bij het creëren en vergroten van het vertrouwen. Het geeft je als gebruiker een bepaalde verantwoordelijkheid, waardoor er vertrouwen ontstaat tijdens het leerproces. Als je daarbij geen perfectie van het systeem verwacht of nastreeft, werkt het best goed. Denk hierbij aan spraakherkenning op de smartphone, zoals genoemd in hoofdstuk 1. Het systeem kent mij en mijn mailgedrag waardoor ik het ga vertrouwen. Op een bepaald punt is het niet meer nodig om continu alles voor te leggen, alleen nog bij twijfel. Het systeem wordt een collega: "Je gaat zo'n systeem personaliseren. Het wordt een hij of zij. Dit is ook een uiting van vertrouwen." Het vertrouwen hoeft niet optimaal te zijn volgens een enkeling: "Nee en dat vind ik ook niet nodig. Want je moet ook een beetje afstand houden tussen jou en de machine. Het is geen gezelschapsrobot"

Suggesties van de deelnemers voor het creëren van meer vertrouwen:
- Het model trainen met meer gegevens en met meer mensen.
- Het inbouwen van menselijke controle (steekproefsgewijs controleren). Als je dit kunt inbouwen in de trainingsloop, dan ontstaat er een feedback naar het systeem.
- Nadenken hoe je het classificatiemodel gaat onderhouden en dit teruggeven aan de gebruiker.
- Een combinatie met andere technieken vergroot het vertrouwen. Specifiek voor e-mail kun je kijken naar de headers (de gestructureerde informatie) en de bijgevoegde attachments.
- Nadenken wat je wilt met de classificatie. Het zijn nu enkel aanbevelingen. Als je er acties aan verbindt, moet dit wel duidelijk zijn.

3.2.5 Toepasbaarheid van het prototype

Op de vraag of de deelnemers een toepassing als deze zouden gebruiken – mits verder ontwikkeld – was het antwoord een unaniem JA.

Niemand heeft er moeite mee als het model na een bepaalde periode een actie koppelt aan een bepaalde klasse, bijvoorbeeld het verwijderen van ruis uit de e-mailbox van de gebruiker. Voorwaarden hiervoor zijn wel dat er goed nagedacht moet zijn over wanneer dit plaatsvindt, en er moet een controle dan wel interventiemogelijkheid aanwezig zijn. Je kunt beginnen met het koppelen van een actie aan die voorspellingen waarvan je (bijna) zeker weet dat deze correct zijn. Bijvoorbeeld voorspellingen met een zekerheid (zie hfst. 2.3.3) van 80% en hoger. "Geautomatiseerd weggooien kan een goed hulpmiddel zijn voor organisaties. Op dit moment besteden we al veel te veel tijd aan het afhandelen en verwerken van e-mail. Als we deze tijd kunnen verkleinen en opslagruimte kunnen besparen, betekent dit al winst."

Het zelf trainen van een (classificatie)model creëert draagvlak voor een nieuwe technologie. Als je dit inzet voor een grote groep medewerkers, dan is het misschien niet nodig om elk e-mailbericht te controleren. Je moet op zoek naar een basis, zoals een classificatiemodel dat al tot op zekere hoogte is getraind. De mogelijkheid tot eigen vorming moet wel aanwezig blijven. Anders gezegd: de 'slimme dingen' wil je generiek regelen. Details worden steeds beter naarmate het model jou gaat herkennen. Als er vertrouwen is in het systeem, hoef je het maar eens in de zoveel tijd te controleren.

Hieronder staan aandachtspunten voor doorontwikkeling, die deelnemers aan het experiment hebben aangegeven. Deze hebben overlap met de suggesties in hoofdstuk 3.2.4, wat aantoont dat vertrouwen een belangrijke stap is bij het verder ontwikkelen.

- Het inbouwen van controles. Er moet een mogelijkheid zijn om de beoordeling aan te passen.
- Nadenken wat belangrijk is binnen je organisatie en het opstellen van business rules. Een generiek organisatiemodel en persoonlijke 'fine tuning' mogelijkheden.
- Het systeem moet zelfstandig werken en 'realtime' classificaties geven. Het systeem moet op de achtergrond aanwezig zijn en zoveel mogelijk geïntegreerd zijn in de huidige kantoorapplicaties.
- Combinaties realiseren met andere strategieën en technologieën. Denk aan zoek- en analysetechnologie en regels maken aan de hand van gestructureerde informatie van de e-mailberichten. Idealiter is het classificatiemodel een onderdeel van een grotere toolkit.

De volgende opmerking van een deelnemer is de kant die we op zouden moeten:

> *"Gebruik de voordelen van een machine: snel patronen herkennen. Een machine heeft patronen eerder door dan wij. Én de mens. Die weet direct aan de bron… hier begint iets. Dit zijn sleutelfiguren of dit zijn woorden die bij een bepaald thema horen."*

3.3 Samenvattend

De feitelijke cijfers tonen aan dat er veel 'rommel' in onze mailboxen zit. Aan het begin van het experiment hebben we de inschatting gemaakt dat we waarschijnlijk 40 tot 50% van de e-mail weg kunnen gooien. Deze aanname klopte. Alleen al door ruis mail aan te merken en te verwijderen uit 'de grote bak' kunnen we tot bijna de helft aan opslag besparen. En dan hebben we het nog niet over de categorisering van functioneel belangrijk en functioneel niet-belangrijk.

De afzonderlijke algoritmen gaven verschillende resultaten. Gebaseerd op dit prototype waren de twee forest algoritmen het meest stabiel. Het Multinominal Naive Bayes algoritme had een grotere schommeling. Het algoritme stabiliseerde zich wel en werd in de loop van het experiment beter. Het lijkt erop dat voor dit algoritme grotere hoeveelheden van trainingsdata nodig zijn, voordat kenmerken zich gaan onderscheiden. Het algemene classificatiemodel werd gedurende de trainingsperiode 'slimmer'. Het werd goed in het herkennen van duidelijke ruis e-mails en duidelijke functionele e-mails. E-mails met zowel ruis als functionele boodschappen en de twijfelgevallen herkende het model moeilijker. Dit was logisch. Medewerkers gaven aan dat zij het zelf ook lastig vonden om deze e-mailberichten te beoordelen.

Vertrouwen in een zelflerend systeem is anders dan we in eerste instantie dachten.
Controle of in ieder geval het gevoel van controle hebben over het classificatiemodel, lijkt belangrijker dan inzicht hebben in het feitelijk functioneren van de algoritmen. Voor nu is de mogelijkheid om het algoritme te kunnen corrigeren nog één van de belangrijkste aspecten voor vertrouwen in het systeem. Daar komt bij dat de medewerker een verantwoordelijkheid voelt voor het goed functioneren van het systeem, omdat hij of zij zelf de trainer is. Je wilt het systeem zo goed mogelijk trainen. Door deze interactie wordt de eigen (menselijke) inconsequentie veel zichtbaarder. Daarnaast wordt de medewerker zich ervan bewust dat een perfect systeem niet bestaat en niet nodig is. Het gevoel dat we het (nog) kunnen controleren is een sterke menselijke eigenschap. Deze moeten we niet onderkennen bij de acceptatie van deze systemen.

Vertrouwen in de persoon (of personen) achter het systeem draagt ook in grote mate bij aan de acceptatie; meer dan we in eerste instantie dachten. Een goede uitleg geven is nodig: hoe werkt het systeem, wat doe je ermee (welke vraag moet het beantwoorden) en met welk doel (waarom zoeken we een antwoord op die vraag)? Goed en nauwkeurig kunnen uitleggen hoe je als organisatie omgaat met de gegevens van de medewerkers is essentieel.

Het detailscherm ontwikkelden we om de medewerker inzicht te geven in het functioneren van de algoritmen. Daarmee hoopten we het vertrouwen in het systeem te vergroten. De deelnemers hebben nu niet of nauwelijks gekeken op deze schermen. De vraag is of het scherm echt overbodig was. Wat als het scherm er niet was geweest? Met andere woorden: nu hadden de medewerkers de keuze om wel of niet detailinformatie te bekijken. Wat als de keuze er niet was, zouden medewerkers dan aangegeven hebben dat ze dit misten? Zouden ze minder vertrouwen gehad hebben in het systeem? Dit is misschien iets om bij een vervolg mee te nemen.

4. Bevindingen

Aan het einde van hoofdstuk 2 en 3 hebben we de belangrijkste lessen van het experiment Machine Learning en Automatische Classificatie samengevat. In dit hoofdstuk vertalen we dit naar meer algemene bevindingen over de toepasbaarheid van machine learning. Daarnaast beantwoorden we de vragen die we aan het begin van het experiment stelden.

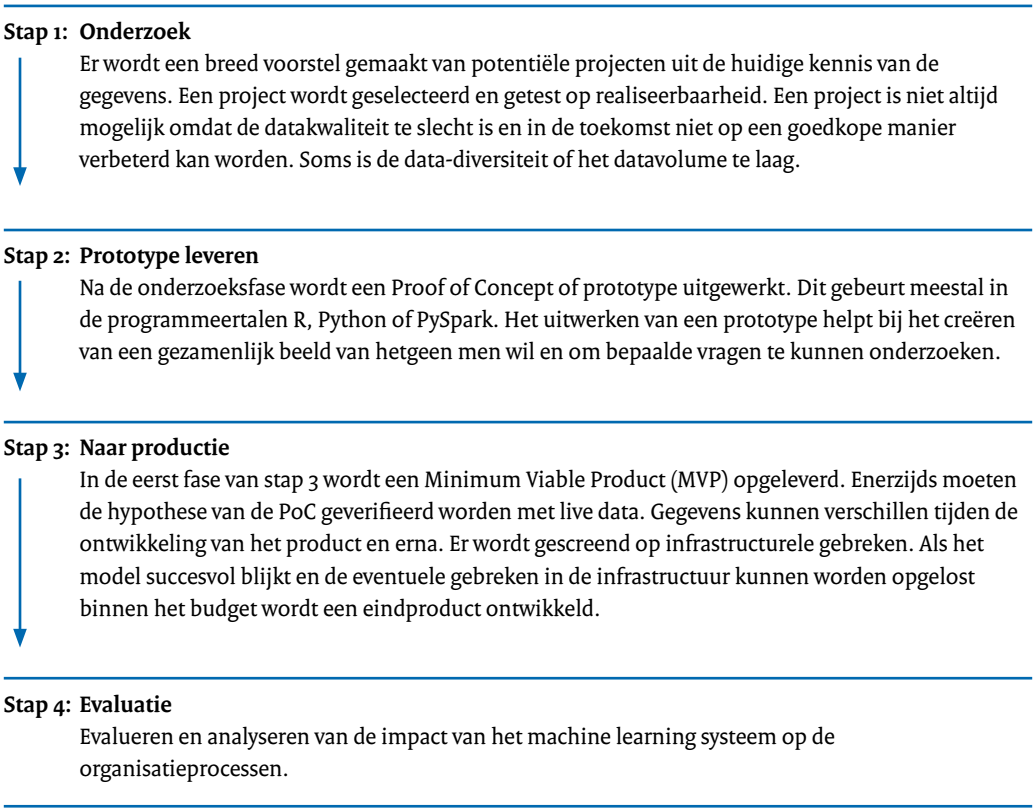
4.1 Machine learning en de toepasbaarheid

Ondanks de beschikbaarheid van open source pakketten hebben we ondervonden dat er behoorlijk wat werk nodig is om de gewenste resultaten te bereiken. Dit geldt niet alleen voor het inzetten van machine learning technologie, maar ook voor het uitvoeren van een experiment binnen je eigen organisatie. Je moet je weg vinden in een bijna oneindig aantal mogelijkheden en oplossingen. Daarnaast bestaat een zelflerend systeem uit verschillende componenten, die alle bijdragen aan het functioneren van het systeem en de kwaliteit van de uiteindelijke voorspelling. Deze componenten moeten we goed op elkaar afstemmen en inpassen in de infrastructuur van onze organisatie.

Commerciële bedrijven bieden platforms aan die gebruik maken van machine learning voor classificatie-mogelijkheden. Hierdoor ontstaat het beeld dat de technologie niet meer is dan een stukje software, waar je een licentie voor koopt of die je kunt installeren en dan direct werkt. Maar achter deze platforms draait een uitgebreide infrastructuur bij de bedrijven zelf, die het zelflerende component mogelijk maakt. Hoe de platforms de gegevens precies verwerken, welke algoritmen ze gebruiken en/of met welke features ze trainen, blijft voor nu onduidelijk. Als je in zee gaat met een commerciële partij, verdiep je dan eerst in wat deze bedrijven echt aanbieden en wat jou als organisatie kan helpen. Kunstmatige Intelligentie en machine learning zijn namelijk buzz-woorden en technologische containerbegrippen die helpen een toepassing te verkopen. Laat je vooraf goed informeren.

4.2 Een machine learning project draaien

Machine learning en het toepassen van big data experimenten zijn de laatste jaren op gang gekomen binnen verschillende sectoren. Ook in Nederland is het vakgebied de afgelopen jaren toegenomen. Uit projecten is een standaard aanpak (zie fig.20) voortgekomen met daarbinnen specifieke stappen en rollen. Tijdens de ontwikkelfase combineerden we deze aanpak met een agile aanpak om te kunnen experimenteren.



Figuur 20 - Standaard aanpak van een machine learning project

Het experiment heeft onderdelen van stap 1, stap 2 en stap 3 gerealiseerd. Het is goed om te weten dat er een standaard aanpak is voor dit soort projecten. Het is bovendien belangrijk dat je er een eigen invulling aan geeft.

Een relevant verschil om te noemen is het verschil in rollen tussen een data scientist en een data engineer. Data scientists voeren statistische analyses uit om te onderzoeken welke machine learning aanpak een organisatie het beste kan gebruiken en welk algoritme ze het beste kunnen inzetten om het probleem op te lossen. Een data scientist ontwikkelt een prototype. Een eventuele uitwerking van een prototype vraagt om een andere rol: de data engineer. Een data engineer gebruikt meer robuustere programmeertalen om een product te ontwikkelen. Het verschil tussen de twee rollen is dat de data scientist meer weet van statistiek en actief is in de beginfase. De machine learning engineer is juist sterker in programmeren en verzorgt de implementatie.

Houdt er rekening mee dat de ontwikkeling van een zelflerend systeem niet alleen technisch is. Er zijn mensen nodig die de inhoud van de gegevens en werkprocessen kennen en duidelijk op de hoogte zijn van de problematiek of het gewenste resultaat. Voor een gedeelte moet je je laten leiden door de gegevens: kunnen ze de vraag beantwoorden die ik stel? Kan dit niet, dan hoeft dit niet te betekenen dat je de gegevens niet kunt gebruiken of je niet op de goede weg bent. Misschien moet je in dat geval een andere vraag stellen.

4.3 Open deuren

Met het lezen van artikelen en de vele theoretische uiteenzettingen over machine learning komen vaak dezelfde bevindingen en waarschuwingen naar voren. Het zijn vaak open deuren, maar daarom niet minder waar. Onderstaande punten hebben we zelf ondervonden in het experiment:

1. **Het perfecte systeem bestaat niet**
   De aanname dat een systeem geen fouten kan of mag maken is hardnekkig. Het liefst zien we een 100% nauwkeurige voorspelling, wat onmogelijk is. Daarbij overschatten we onszelf en het vermogen van onze eigen nauwkeurigheid. Een systeem verwerkt grote hoeveelheden aan informatie consequenter dan wij zelf kunnen.

   In hoofdstuk 3 hebben we gezien dat zelf trainen en de interactie met het systeem ons veel leert over ons eigen gedrag. Het systeem confronteert ons niet alleen met onze eigen inconsequentie, maar we gaan ook op een andere manier om met onze e-mail. Vertrouwen is iets dat je moet opbouwen. Dat lukt alleen als je niet streeft naar perfectie.

2. **Garbage in is garbage out**
   Gegevens moeten een goede kwaliteit hebben. Algoritmen werken beter op 'schone' gegevens. Voor het toepassen van zelflerende algoritmen zijn ook nog eens grote hoeveelheden gegevens nodig.
   Wat betekent dit? Haal in ieder geval de gegevens eruit die niet relevant zijn voor je eigenlijke vraag. Een investering vooraf komt uiteindelijk ten goede aan de training en nauwkeurigheid van je systeem.

   Het filteren van ruis, zoals toegepast in dit experiment, is de eerste stap in een groter geheel. Je kunt na deze stap op verschillende manieren verder. Na het filteren van ruis heb je een bak met functionele e-mailberichten over. Wat wil ik weten van deze gegevensverzameling? Met unsupervised machine learning kun je kijken of algoritmen patronen in de gegevens herkennen die je van tevoren niet kunt bedenken. Of denk na over het inzetten van supervised machine learning voor bijvoorbeeld het beoordelen van openbaarheid (zie uitgebreid Bijlage A).

3. **De juiste vraag is cruciaal**
   De gegevens die je gebruikt voor het trainen van het model, moeten de vraag die je aan de gegevens stelt kunnen beantwoorden. Met andere woorden: wat wil je weten van de gegevens? Als de gegevens je niets kunnen vertellen over de vraag die je probeert te beantwoorden, dan gaat zelfs het meest krachtige lerende algoritme dat niet voor je doen.

4. **Gebruik van nieuwe technologie vraagt om flexibiliteit**
   Omgaan met een enorme hoeveelheid aan digitale informatie vraagt om een andere manier van werken. Anders werken vraagt om flexibiliteit van een organisatie en van zijn medewerkers. Uiteindelijk is het inzetten van zelflerende systemen geen technologisch vraagstuk, maar een verander- en organisatievraagstuk. Deze verandering kan alleen tot stand komen door te experimenteren, mislukken en opnieuw beginnen. Laat gevoelige issues als gegevensverwerking en privacy je niet tegenwerken. Ga de uitdaging aan.

Experimenteren kun je niet alleen. Het vraagt om tijd en mensen vrijmaken en om een structuur om binnen te experimenteren. Betrek medewerkers bij de ontwikkeling van nieuwe tools. We hebben daarnaast gezien dat zelf trainen bijdraagt aan het vertrouwen. Een systeem hoeft niet in één keer af te zijn. Als medewerkers zien waar het naartoe kan groeien en het idee hebben dat ze daaraan kunnen bijdragen (en er op een bepaalde manier controle over hebben), dan accepteren ze het beter en eerder.

4.4 Antwoorden

Heeft dit experiment ons antwoorden gegeven? Buiten de vele nieuwe vragen en ideeën was het een nuttig en succesvol experiment. Laten we eindigen met de vragen die we aan het begin van het experiment stelden.

- **Kunnen zelflerende systemen bijdragen aan een betere informatiehuishouding?**
  Ja. Zelflerende systemen kunnen bijdragen aan een betere informatiehuishouding. Hiervoor moeten we kennis opdoen en ontwikkelen. De ontwikkelingen in zelflerende systemen gaan snel. Als we invloed willen hebben op de ontwikkeling van deze systemen, dan moeten we nu inspringen. Door het beantwoorden van een eenvoudige vraag en/of het opzetten van een klein experiment, kun je als organisatie beginnen beslissingen in te bouwen. Ontwikkel gevoel voor de technologie en de (on)mogelijkheden. Zoek de samenwerking op en bouw voort op andere experimenten en ontwikkelingen die gaande zijn.

- **Is het mogelijk zelflerende systemen in te zetten voor het waarderen, selecteren en toegankelijk maken van ongestructureerde informatie binnen overheidsprocessen?**
  Ja. Met classificatiemodellen kunnen we informatie vroeg in de keten waarderen of labelen. Dit draagt bij aan het selecteren en beter toegankelijk maken van informatie later in de keten. De vraag die je deze systemen laat beantwoorden is cruciaal (dat is de vraag die je stelt aan de informatie). We moeten goed nadenken over wat de mens goed kan en wat de machine goed kan en daarvan profiteren. Stel een duidelijke vraag met een duidelijk antwoord. Zoals we gezien hebben, is het behoorlijk lastig om een systeem te trainen als wijzelf de vraag al niet goed kunnen beantwoorden.

  Een voorbeeld hiervan is onderverdelen in te veel categorieën. Want wat doe je als de informatie in meerdere categorieën kan vallen? Het risico dat we hier lopen, is dat we proberen onze huidige werkwijzen na te bootsen met technologie. Deze zijn gebaseerd op analoge principes. De digitale tijden en zelflerende systemen vragen om een andere manier van werken en benaderen. Dit verzinnen we niet ineens en we hebben niet gelijk het perfecte antwoord voorhanden.

- **Kunnen algoritmen informatie identificeren en toewijzen aan een bepaalde klasse?**
  Ja. Algoritmen kunnen vrij snel en zonder al te veel regels beslissingen van mensen nabootsen. Als je dit bewezen hebt, begint het echte werk pas. Hoe ontwikkel je een stabiel model, hoe kan het draaien binnen de organisatie, hoe regel ik voldoende feedback, hoe gaat het model om met verandering, enz. Er zijn nog genoeg vragen over die we moeten beantwoorden.

- **Is de technologie al volwassen genoeg om in te zetten in een werkproces?**
  Ja en nee. Als je plug-and-play en one-size-fits all oplossingen verwacht, dan kom je niet ver. Zoals we hebben gezien in hoofdstuk 4.2, is er behoorlijk wat werk nodig. Dit betekent niet dat we het niet kunnen inzetten in een werkproces.

- **Wat is nodig om vertrouwen te krijgen in de beslissingen die deze systemen voor ons maken?**
  Vertrouwen lijkt op dit moment meer te maken te hebben met controle. Dat wil zeggen het idee dat je controle en invloed kunt uitoefenen op de resultaten van het systeem. Vertrouwen in het systeem heeft te maken met vertrouwen in 'de persoon' achter het systeem. Kan diegene uitleg geven over de doeleinden van classificatie, over hoe het systeem is opgebouwd, met welke kenmerken er getraind wordt, gaat de organisatie nauwkeurig met de gegevens om, enz. Welk algoritme de systemengebruiken en hoe deze feitelijk presteren, is of lijkt op dit moment veel minder belangrijk.

Bronnenlijst

1. Het idee dat de gebruiker zelf het algoritme traint is mede ontstaan vanuit de juridische beperkingen m.b.t. privacy (zie maatregelen Hfst 2.2.1).
   Toon bron in tekst
2. Het onderzoeksplan opgesteld in samenwerking met de Hogeschool van Amsterdam voor subsidieaanvraag bij RAAK publiek (ID-proza 919496) ligt ten grondslag aan dit project. Na afwijzing van de subsidieaanvraag heeft het Nationaal Archief besloten het onderzoek toch verder vorm te geven, waarbij de scope en omvang is aangepast (ID-proza 920080).
   Toon bron in tekst
3. Het Discovery proces is een onderzoek, voorafgaande aan een rechtszaak, waar elke partij bewijsmateriaal van de andere partij kan verkrijgen. Het kan gaan om ondervraging, een verzoek tot het leveren van bepaalde informatie en/of documenten. Dit gebeurt door het indienen van een (informatie) verzoek. E-Discovery (of Electronic Discovery) kan omschreven worden als een proces dat helpt bij het doorzoeken van grote hoeveelheden elektronische gegevens en/of informatie voor een specifiek doeleinde. Dit is vaak een juridisch onderzoek of rechtszaak.
   Toon bron in tekst
4. Beknopte uitleg van de publicatie is te vinden op wikipedia https://en.wikipedia.org/wiki/Computing_Machinery_and_Intelligence. Publicatie: https://www.csee.umbc.edu/courses/471/papers/turing.pdf.
   Toon bron in tekst
5. Beschrijving van The Imitation Game: Suppose that we have a person, a machine, and an interrogator. The interrogator is in a room separated from the other person and the machine. The object of the game is for the interrogator to determine which of the other two is the person, and which is the machine. The interrogator knows the other person and the machine by the labels 'X' and 'Y'—but, at least at the beginning of the game, does not know which of the other person and the machine is 'X'—and at the end of the game says either 'X is the person and Y is the machine' or 'X is the machine and Y is the person'. The interrogator is allowed to put questions to the person and the machine of the following kind: "Will X please tell me whether X plays chess?" Whichever of the machine and the other person is X must answer questions that are addressed to X. The object of the machine is to try to cause the interrogator to mistakenly conclude that the machine is the other person; the object of the other person is to try to help the interrogator to correctly identify the machine. https://plato.stanford.edu/entries/turing-test/
   Toon bron in tekst
6. ML is the subfield of computer science that " gives computers the ability to learn without being explicitly programmed" (Arthur Samuel, 1959).
   Toon bron in tekst
7. Machine learning for videogames https://www.youtube.com/watch?v=qv6UVOQ0F44
   Toon bron in tekst
8. Waardering van informatie gebeurt ten behoeve van verschillende doeleinden: bijvoorbeeld om aan te geven welke informatie gepubliceerd kan worden of welke informatie voorlopig niet in de openbaarheid mag komen en welke informatie optimaal beveiligd dient te worden. Een ander oogmerk van waarderen is om te bepalen welke informatie hoe lang bewaard dient te worden.
   Toon bron in tekst
9. Bron: Bewaren van e-mail Rijksoverheid – concept handreiking (https://www.informatiehuishouding.nl/documenten/beleidsnotas/2016/12/2/white-paper-archiveren-van-e-mail)
   Toon bron in tekst
10. Uit een steekproef bij een ministerie blijkt dat slechts 2,5% van de e-mail (3,6 miljoen e-mails) wordt opgeslagen in hun DMS
    Toon bron in tekst
11. https://www.rijkaaninformatie.nl/projecten/e-mailarchivering
    Toon bron in tekst

12. Beschreven in de Whitepaper Archiveren van e-mail in de Rijksoverheid https://www.rijkaaninformatie.nl/documenten/beleidsnotas/2016/12/2/white-paper-archiveren-van-e-mail
Toon bron in tekst
13. Lees de gehele werkwijze in de concept handreiking Bewaren van e-mail Rijksoverheid https://www.rijkaaninformatie.nl/documenten/rapporten/2018/10/12/concept-handreiking-bewaren-van-e-mail-rijksoverheid
Toon bron in tekst
14. Het idee dat de gebruiker zelf het algoritme traint is mede ontstaan vanuit de juridische beperkingen over privacy (zie maatregelen hfst 2.2.2)
Toon bron in tekst
15. Met partners bedoelen we hier zorgdragers (ministeries, zbo's)
Toon bron in tekst
16. Bij de start van het onderzoek/experiment was dit nog de Wbp. Later is dit overgegaan in de AVG
Toon bron in tekst
17. Eenvoudig gezegd voert een spamfilter een binaire classificatie (ja of nee) taak uit, waarbij rechtmatige (goede of HAM) e-mailberichten worden behandeld als een negatieve (-) instantie en SPAM als een positieve (+) instantie. Deze classificatie vindt steeds vaker automatisch plaats. Hierbij wordt gebruik gemaakt van een statistische benadering of machine learning technieken. Het model of de classifier wat ontwikkeld wordt richt zich op één enkele taak namelijk het uitfilteren van spam.
Toon bron in tekst
18. Groslijst technologie – opgesteld maart 2017 | ID – proza 1156944
Toon bron in tekst
19. Classificatie is het voorspellen van antwoorden Ja/Nee, True/False enz. wat een spamfilter ook doet.
Toon bron in tekst
20. Voorwaarde dat we gebruik maakten van Office 365 wat wij niet hebben draaien. Snelle oplossing hiervoor is experimenteren met de cloudversie.
Toon bron in tekst
21. Denk aan e-mailservices van Gmail, Yahoo! Mail, Windows Live Hotmail and AOL.
Toon bron in tekst
22. Onder een black box verstaan we een apparaat, systeem of object dat bekeken kan worden in termen van in- en output, zonder enige kennis van de interne werking te hebben. De implementatie van deze systemen is 'ondoorzichtig' oftewel zwart.
Toon bron in tekst
23. https://discipl.org/
Toon bron in tekst
24. Term die binnen scrum gebruikt wordt voor een korte ontwikkelperiode.
Toon bron in tekst
25. Voorbeeld van de bewerkers Overeenkomst is te vinden in Proza (ID 535321). Inclusief Bijlage I – verklaring gegevensbescherming en geheimhoudingsverklaring, Bijlage II Passende technische en organisatorische maatregelen en Bijlage III Maatregelen in verband met de meldplicht datalekken.
Toon bron in tekst
26. Overeenkomst deelname experiment medewerkers te vinden in Proza (ID 547868)
Toon bron in tekst
27. https://api.ning.com/files/VGBOPpzwSaK66PQD7Nt-qqZyB1VMOvMkZCMznYpnySPyhV1Cgp77wctaAh4JB48f6106osieLXq6Qj5nKsbTsiBVoDPq9IsP/080701EmailgedragslijnBaselinev1.0.pdf
Toon bron in tekst
28. Adidas koppeling komt van het fysiek heen en weer brengen van e-mailgegevens op een USB stick. Op gymschoenen/sneakers gaat dit een stuk sneller
Toon bron in tekst
29. https://computerworld.nl/cloud/84263-wat-is-docker
Toon bron in tekst

30. Elke medewerker heeft een eigen gebruikersnaam en inlog gekregen. Op basis van deze gegevens worden de e-mailberichten weergegeven. De medewerker krijgt enkel die e-mailberichten te zien waar hij/zij in de “van” of “aan” wordt vermeld
    Toon bron in tekst
31. De e-malberichten met een bevestigd label dienen als nieuwe trainingsset voor de verschillende algoritmen.
    Toon bron in tekst
32. Mogelijke antwoorden: Ja het is ruis = True Positive of Nee het is geen ruis, maar functionele mail = True Negative
    Toon bron in tekst
33. Er is besloten om het ‘slechtst’ functionerende algoritme het meest te trainen
    Toon bron in tekst
34. De daling van 16 juli is niet te verklaren. In deze week was de projectleider afwezig maar het algoitme was wel getraind (waarden in de confusion matrix waren veranderd). De nauwkeurigheid is na deze datum weer omhoog geklommen.
    Toon bron in tekst
35. Dit zijn de e-mailberichten van de projectleider van het experiment en niet de e-mailberichten van de deelnemers. Om deze te gebruiken voor evaluatie is er toestemming van de deelnemer nodig.
    Toon bron in tekst
36. Misschien nog noemen in de ze verwijzing, dat het e-maildetailscherm speciaal is ontwikkeld om vertrouwen en transparantie in de algoritmen op te bouwen.
    Toon bron in tekst
37. De ander kant op is waarschijnlijk minder erg (ruis die als zakelijk wordt geclassificeerd).
    Toon bron in tekst
38. Het gedeelte van de nieuwe werkwijze emailarchivering waarop de stap betrekking heeft.
    Toon bron in tekst

Bijlage A

oplossingsrichting E-Discovery voor informatiemanagement

Stap 2: Toepassen op echte omgeving

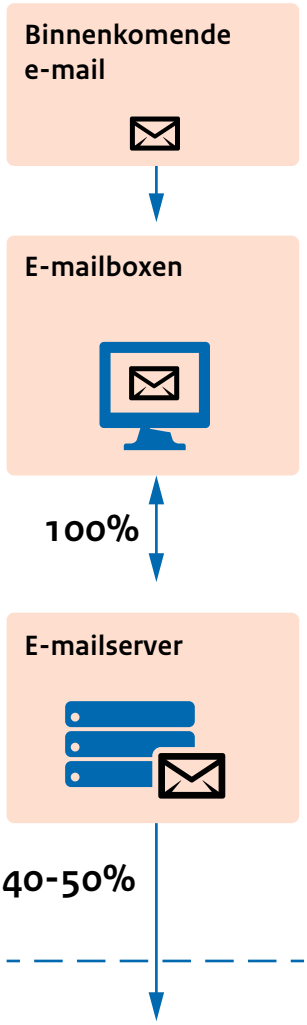
Installatie NA omgeving - verder trainen model

Onderzoeksvraag:
Kan het in stap 1 ontwikkelde computermodel ingezet worden op e-mailboxen en binnen-komende e-mailberichten van (andere) medewerkers en binnen andere organisaties

Stap 1: Filteren van ruis

PROTOTYPE ontwikkeld in lab-omgeving

Onderzoeksvraag:
Hoe kunnen we technologie inzetten om de e-mailverzamelingen die ontstaan te ontdoen van ruis, waardoor de hoeveelheid van (niet-taak gerelateerde) informatie die we opslaan afneemt.

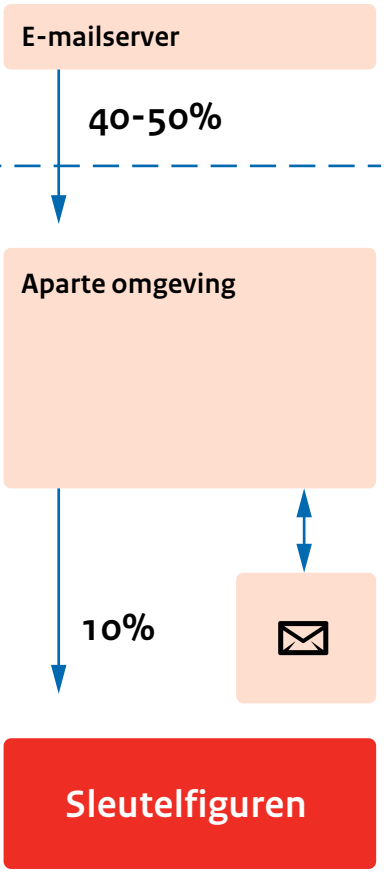


Stap 1 – Filteren van Ruis
Werkwijze[38]: Medewerkers kunnen in de eerste tien weken privé e-mail, personeelsvertrouwelijke zaken of andere, niet archiefwaardige e-mail, verwijderen.

De medewerker gaat dit niet doen (als je het niet gelijk doet) en zal geholpen zijn als de technologie hem of haar hierbij ondersteunt.

Stap 2 – Toepassen in een echte omgeving
Werkwijze: Medewerkers kunnen in de eerste tien weken privé e-mail, personeelsvertrouwelijke zaken of andere, niet archiefwaardige e-mail, verwijderen.

Aanname: Als medewerkers inkomende en uitgaande e-mail direct labelen, is ongeveer de helft 'ruis'.
Kan het in stap 1 ontwikkelde computermodel ingezet worden op e-mailboxen en binnenkomende e-mailberichten van (andere) medewerkers en binnen andere organisaties?



Stap 3: Automatische classificatie

Toepassen van Technology Assisted Review (TAR) oplossingen

Onderzoeksvraag:
Hoe kunnen zoek- en analysetechnologie, gebruikt binnen het vakgebied van E-Discovery, worden ingezet voor automatisch waarderen en selecteren van informatie zodat er beter voldaan kan worden aan de informatiebehoefte van de gebruiker van informatie.

Stap 4: Beoordelen openbaarheid

Verder uitwerken ML mogelijkheden

Onderzoeksvraag:
Hoe kan predictive coding worden ingezet om de gebruiker van de informatie te ondersteunen bij het beoordelen op openbaarheidsbeperkingen geldende onder de wob en archiefwet.

Stap 3 - Automatische classificatie
Werkwijze: E-mails van en naar medewerkers van de Rijksoverheid wordt – tien weken na verzending of ontvangst – automatisch in een aparte omgeving opgeslagen. Daar blijft het tien jaar staan.

Aanname: als ruis is verwijderd, zullen zoek- en analysetechnieken betere resultaten opleveren en algoritmen beter werken.
Hoe kunnen we zoek- en analysetechnologie van het vakgebied E-Discovery inzetten om informatie automatisch te waarderen en selecteren, zodat we beter voldoen aan de informatiebehoefte van de gebruiker?

Stap 4 - Beoordelen openbaarheid
Beschrijving: Voor te benoemen sleutelfunctionarissen wordt de e-mail permanent bewaard. Na uiterlijk twintig jaar, of zoveel eerder als ministeries beslissen, gaan de berichten over naar het Nationaal Archief.

Aanname: Hoe zetten we predictive coding in om gebruikers van informatie te ondersteunen bij het beoordelen op openbaarheidsbeperkingen, geldend onder de WOB en de Archiefwet?

Een uitgave van het Nationaal Archief
Postbus 90520
2509 LM Den Haag

Contact
Telefoon: +31-70-331 5460
E-mail: contact@nationaalarchief.nl

Website: https://www.nationaalarchief.nl/archiveren