De kern van een AI-systeem is het neurale netwerk, die op basis van input en voorgegeven output (trainingsdata) zichzelf kan configureren om bij vergelijkbare input de juiste output te geven. Dit heet “machine learning”.

Hier zijn risico’s aan verbonden: als de trainingsdata onvolledig is, of disproportioneel wordt beïnvloed door bijvoorbeeld klikgedrag van gebruikers, kan de output ook onjuist zijn. Dit gebeurde bijvoorbeeld in Myanmar waar het facebookalgoritme gebruikers een steeds negatiever beeld gaf van de Rohingya, een minderheid. Het algoritme had daardoor een onbedoeld akelige rol in etnische zuiveringen op de Rohingya.

Bias

Deze risico’s worden wel samengevat onder de term “bias” (vooringenomenheid, verstelling). Er zijn verschillende vormen van bias, zoals “underfitting”, waarbij de trainingsdata te beperkt is en daardoor het beeld van de werkelijkheid te simpel is. Een ander voorbeeld is “selection bias” waarbij de trainingsdata al onbedoeld een onterechte voorselectie bevatten of missen. Een voorbeeld deed zich voor bij de laatste Europese verkiezingen: de polls vergaten dat vooral de eurosceptische kiezers vaker niet gingen stemmen dan anderen. Eurosceptische partijen deden het daardoor in de verkiezing slechter dan in de polls.

Bij AI-systemen is het algoritme (het uiteindelijke gedrag, de beslissingscriteria) niet vastgelegd in de code maar het resultaat van de code en vooral ook van de trainingsdata. Hierdoor kun je het systeem niet gericht fixen zonder dat dit voor het gehele algoritme gevolgen heeft.

De risico’s en het karakter van AI zorgen er voor dat de testvraag bij AI dynamisch wordt: je wilt bij AI weten hoe het gehele systeem presteert voor alle partijen. Hiervoor heeft het Polteq AI team een tool ontwikkeld, de **Increment Quotient tool (iQ)**, waarmee testgevallen en metrieken en een overall evaluatie kunnen worden opgesteld en uitgevoerd in een AI-systeem.

Een AI-testtraject bestaat uit de volgende activiteiten:

1. Analyseren van betrokken partijen (de databronnen, labels, klanten, gebruikers, mensen in de wereld zoals de Rohingya). Hiervoor stel je “persona’s” op;
2. Per persona bepalen wat de verwachtingen bij de gegeven input zijn, bijvoorbeeld wat iemand verwacht als top 1 resultaat in een zoekmachine;
3. Bepalen van metrieken die van belang zijn: wil je bijvoorbeeld weten welk percentage van de top 10 resultaten relevant is of hoe hoog een belangrijk resultaat staat?
4. Onderling belang van de verwachtingen en metrieken;
5. Inrichten van de iQ tool zodat deze per testgeval input genereert voor het AI-systeem en de resultaten afvangt en verwerkt, zodat je een overzicht van alle resultaten hebt en

een totaal score.

6. Per versie van het AI-systeem wordt de iQ tool gedraaid. Eindresultaten per versie zijn zo een indicatie voor een afweging welke versie voorkeur verdient en de iQ tool geeft programmeurs aanwijzingen voor aanpassingen in trainingsdata of parametrisatie zodat het systeem geoptimaliseerd kan worden.

Gerard Numan, Michiel Keij en Mathijs Kadijk (Q42)

Naast het hebben van basiskennis van metrieken en data science en het kunnen uitvoeren van bovenstaande activiteiten moet een AI-tester vooral empathie beoefenen. Dat is misschien wel het belangrijkste. De tester moet zich namelijk inleven in alle partijen: uit gesprekken en onderzoek moet hij of zij profielen, behoeftes, noden en normen boven tafel krijgen en kunnen inschatten wat de gevolgen kunnen zijn. Ook moet de tester sensitief zijn voor sociale en morele context. Al deze verwachtingen moeten worden vertaald in gewogen metrieken. De testresultaten bestaan vooral uit rekenresultaten en deze moeten zinnig worden geëvalueerd en getotaliseerd.

De tester moet begrijpen wat de behoeften en beperkingen van de programmeur zijn: deze kan immers niet meer simpelweg bug fixen en wil vooral weten hoe het systeem als geheel presteert ten opzichte van vorige versies en wat normen zijn die niet mogen worden overschreven.

Gerard Numan, Michiel Keij en Mathijs Kadijk (Q42)