Foutlokalisatie in enquêtedata met behulp van (zachte) regels en machine learning

Het Centraal Bureau voor de Statistiek (CBS) maakt jaarlijks productiestatistieken. Om betrouwbare statistieken te kunnen maken is het van belang dat de data van goede kwaliteit zijn. Er wordt daarom onder andere gewerkt aan het verbeteren van het automatische data-opschoningsproces, zodat fouten in enquêtedata worden gelokaliseerd. Na deze opschoning zitten er nog steeds fouten in de data. Daarom worden de data aanvullend handmatig opgeschoond. Voor mijn afstudeerscriptie (onderdeel van het project EBN2.x) heb ik twee methoden onderzocht voor het lokaliseren van fouten in enquêtedata om het opschoningsproces te verbeteren.

Tania Keijzer

Project EBN2.x heeft als doel de enquêtedata zo vroeg mogelijk consistent te maken, zo veel mogelijk automatisch te corrigeren en het handmatige werk te sturen via kwaliteitsmaten. In dit omvangrijke project wordt onder andere gewerkt aan het verbeteren van het automatische opschoningsproces. Hierbij worden voor de hand liggende fouten gecorrigeerd, foutieve waardes opgespoord en vervangen door geldige waardes met behulp van harde regels. Harde regels zijn gedefinieerde regels waar de data altijd aan moeten voldoen, zoals regels over het domein van variabelen en regels die het verband tussen variabelen aangeven. Na het uitvoeren van het opschoningsproces zitten er nog steeds veel fouten in de enquêtedata die handmatig gecontroleerd moeten worden. Dit kost veel tijd en moeite. Hoe zou dit opschoningsproces verbeterd kunnen worden, zodat er minder handmatige controles uitgevoerd moeten worden?

Er zijn twee mogelijke verbeteringen onderzocht. Ten eerste worden in het huidige opschoningsproces naast de harde regels ook zachte regels opgenomen. Dit zijn regels die wijzen op opvallende waarnemingen die als fout gezien kunnen worden, maar waarbij dit niet altijd het geval hoeft te zijn. Ten tweede wordt gekeken of de fouten gelokaliseerd kunnen worden met behulp van een *machine learning* benadering. Dit onderzoek is toegepast op de enquêtedata van de productiestatistieken. Ik heb de methodes toegepast op bedrijven uit de industriesector met 20 tot 49 werkzame personen.

Mixed integer programmeerprobleem

Per bedrijf worden er meer dan honderd variabelen uitgevraagd. Het is van belang om te achterhalen welke variabelen fout zijn bij elk bedrijf. Door het grote aantal variabelen is het niet evident welke (combinatie van) variabelen ervoor zorgen dat regels geschonden worden. In het huidige opschoningsproces worden foutieve waardes opgespoord met behulp van een mixed integer programmeerprobleem (MIP). De doelfunctie, een gewogen som van het aantal foutieve variabelen, wordt geminimaliseerd waarbij aan alle harde regels moet worden voldaan:

$$D_{HARD} = \sum_{j=1}^{p} w_j y_j$$

waarbij $y_j = 1$ als de variabele x_j als foutief wordt aangewezen, anders $y_j = 0$, w_j is de bijbehorende wegingsfactor en p is het aantal variabelen dat gewijzigd kan worden in de data (De Waal, Pannekoek & Scholtus, 2011). Met dit model worden met de harde regels 89% van de fouten niet gedetecteerd (*recall* = 0,11) en 31% van de voorspelde fouten is officieel geen fout (*precision* = 0,69). Dit levert een f1-score, het harmonisch gemiddelde tussen de recall en precision, van 0,19. In het werkelijke productieproces zijn deze cijfers waarschijnlijk anders dan in dit onderzoek, omdat hier niet de data van het laatste stadium van het opschoningsproces gebruikt kon worden als input van het MIP-model. Na het lokaliseren van de fouten worden geldige waardes ingevuld zodat aan alle harde regels wordt voldaan.

In het kader is een voorbeeld van het MIP-model met de harde regels uitgewerkt. Hierin wordt maar één harde regel meegenomen in het MIP-model. In werkelijkheid zijn er honderden harde regels bij de productiestatistieken, wat het probleem zo complex maakt.

Stel de variabelen 'winst' ($x1$), 'omzet' ($x2$) en 'kosten' ($x3$) hebben respectievelijk de wegingsfactoren 2 ($w1$), 4 ($w2$) en 3 ($w3$), de harde regel 'winst is omzet minus kosten' geldt en de volgende dataset geeft per rij de ingevulde enquête van een bedrijf weer:

Bedrijf	Winst	Omzet	Kosten
1	800	1000	300
2	300	500	200

De bovenste rij schendt de harde regel. Daarom moet minstens één van deze variabelen gewijzigd worden om aan de harde regel te voldoen. De variabele 'winst' wordt als fout aangemerkt en vervangen door een missende waarde, omdat deze variabele de laagste wegingsfactor heeft. Op deze manier wordt de doelfunctie geminimaliseerd. In het vervolg worden de missende waardes vervangen door geldige waardes. Bij dit probleem zou eventueel de volgende zachte regel geformuleerd kunnen worden: 'winst mag niet groter zijn dan 50% van de omzet'.

Mixed integer programmeerprobleem met de zachte regels

Door de zachte regels toe te voegen aan het MIP-model wordt de doelfunctie gewijzigd naar het minimaliseren van de som van het aantal foutieve variabelen en het aantal zachte regels dat wordt geschonden (Scholtus, 2013). Hierbij geldt nog steeds de voorwaarde dat aan alle harde regels moet worden voldaan.

$$D = D_{HARD} + D_{ZACHT}$$

$$D_{ZACHT} = \sum_{k=1}^{K_s} S_k z_k$$

waarbij z_k=1 als de zachte regel k niet voldoet, anders z_k=0, S_k is de bijbehorende wegingsfactor en K_s is het aantal zachte regels. Bij het minimaliseren van deze doelfunctie wordt dus een afweging gemaakt tussen de wens om zo weinig mogelijk variabelen te wijzigen en de wens om aan zo veel mogelijk zachte regels te voldoen. Het aantal zachte regels dat wordt meegenomen in het MIP-model is heel flexibel, waarbij de rekentijd wel groter wordt als er meer regels worden meegenomen.

In dit onderzoek zijn met verschillende technieken empirisch zachte regels geformuleerd in de vorm van een lineaire ongelijkheid. Hierbij is enkel een selectie aan zachte regels meegenomen in het MIP-model. Uit de evaluatie blijkt dat toevoegen van de zachte regels aan het huidige MIP-model ervoor zorgt dat er meer fouten worden gelokaliseerd, maar dat er ook meer locaties onterecht als fout worden aangewezen. Met de MIP-modellen met de zachte regels worden maximaal 3% meer fouten gelokaliseerd dan het huidige MIP-model, wat een relatief kleine verbetering is.

Machine learning benadering

Als alternatief voor het toevoegen van zachte regels aan het MIP-model worden verschillende machine learning benaderingen toegepast. De fouten moeten in meerdere variabelen gelokaliseerd worden, waardoor het een multi-label classificatieprobleem is. Het machine learning model multi-label K-Nearest Neighbour (MLKNN), dat met een multi-label dataset kan omgaan, is toegepast.

MLKNN is een uitbreiding van het K-Nearest

Neighbour model (Zhang & Zhou, 2005). In de multi-label dataset heeft elk bedrijf niet één maar meerdere labels, één voor elke variabele. Elk label bestaat uit twee klassen. De klasse '1' betekent dat een variabele voor een bedrijf fout is, anders de klasse '0'. Allereerst worden voor een nieuw bedrijf de K dichtstbijzijnde buren bepaald met de Euclidische afstand. Vervolgens wordt voor elke variabele de kans bepaald dat deze fout is. Daarna wordt het maximum posteriori principe gebruikt om de labels van het nieuwe bedrijf te bepalen.

Daarnaast zijn de data eerst getransformeerd naar meerdere single-label classificatieproblemen, zodat vervolgens de machine learning modellen Naive Bayes en Extreme Gradient Boosting toegepast kunnen worden. Na het uitvoeren van alle machine learning modellen geldt dat nog niet aan alle harde regels wordt voldaan. Om deze reden wordt na een machine learning model altijd het huidige MIP-model uitgevoerd om waar nodig meer fouten aan te wijzen.

Elk machine learning model heeft zijn eigen parameters die afgesteld kunnen worden, bijvoorbeeld het aantal dichtstbijzijnde buren bij MLKNN. Na het afstellen van de juiste parameters door middel van kruisvalidatie geeft MLKNN voorafgaand aan het huidige MIP-model de hoogste recall (0,50). Daarentegen behaalt het machine learning model Extreme Gradient Boosting voorafgaand aan het huidige MIP-model de hoogste f1-score (0,48). Het machine learning model MLKNN behaalt niet de hoogste f1-score, omdat dit model veel meer locaties onterecht als fout aanwijst, waardoor de precision een stuk lager is. Met de machine learning benadering worden maximaal 39% meer fouten gelokaliseerd dan het huidige MIP-model.

Conclusie en aanbevelingen

Tijdens dit onderzoek zijn meerdere modellen onderzocht die tot een verbetering van het huidige MIP-model leiden. Het uitvoeren van de machine learning benadering leidt tot de grootste verbetering. Het machine learning model MLKNN voorafgaand aan het huidige MIP-model lokaliseert veel meer fouten dan het huidige MIP-model, maar wijst ook veel meer locaties onterecht als fout aan. Daarom wordt aanbevolen eerst te onderzoeken of de locaties die onterecht als fout zijn aangewezen nauwkeurig vervangen kunnen worden door geldige waardes voordat dit model in de praktijk wordt toegepast.

Een alternatief is het machine learning model Extreme Gradient Boosting voorafgaand aan het huidige MIP-model. Hiermee worden nog steeds veel fouten gelokaliseerd, maar er worden minder locaties onterecht als fout aangewezen ten opzichte van MLKNN. Er geldt wel dat nog steeds meer locaties onterecht als fout worden aangewezen in vergelijking met het huidige MIP-model. Daarom kan overwogen worden om in een vervolgonderzoek de methodes voor het invullen van geldige waardes te verbeteren als de locaties niet nauwkeurig vervangen kunnen worden door geldige waardes.

Discussie

In dit onderzoek is alleen een voorspelling gemaakt voor bedrijven uit de industriesector met 20 tot 49 werkzame personen. Stel in de toekomst worden de zachte regels in het huidige opschoningsproces meegenomen, dan moet gekeken worden welke zachte regels voor bedrijven in andere sectoren opgesteld kunnen worden. Er worden namelijk verschillende versies van enquêtes uitgevraagd voor de productiestatistieken. Voor het toepassen van een machine learning benadering in de praktijk moet gekeken worden op welke data de algoritmen getraind kunnen worden voor de voorspelling van bedrijven uit andere sectoren.

In dit onderzoek is aangenomen dat een model beter presteert als er meer fouten gelokaliseerd worden, een verhoging van de recall, zonder dat dit voor een extreme verlaging van de precision zorgt. Daarom moet de f1-score ook zo hoog mogelijk zijn. Echter, een model kan ook verbeteren als er minder locaties onterecht als fout worden aangewezen, een verhoging van de precision. De eindpresentatie die ik over mijn onderzoek gaf, eindigde daarom met de vraag: 'Wat is beter: een hogere *recall* of *precision*?'.

Literatuur

De Waal, T., Pannekoek, J., & Scholtus, S. (2011). *Handbook of Statistical Data Editing and Imputation*. Wiley, Hoboken.

Zhang, M.-L., & Zhou, Z-H. (2005). A k-nearest neighbor based algorithm for multi-label classification. *IEEE International Conference on Granular Computing*, Vol. 2, 718–721, doi10.1109/grc.2005.1547385.

Scholtus, S. (2013). Automatic editing with hard and soft edits. *Survey Methodology, 39*(1), 59–89.

Tania Keijzer is afgestudeerd in de Toegepaste Wiskunde aan de Haagse Hogeschool. Voor haar afstudeerscriptie bij de afdeling Methodologie van het CBS heeft ze onderzoek gedaan naar het lokaliseren van fouten in enquêtedata. Haar scriptie is beoordeeld met een 8,5.
E-mail: taniakeijzer@gmail.com

Henk Tijms

column

Foto: Alejandro Garay via Pixabay

DE DODELIJKE GLAZEN BRUG

De Squid Game is één van de meest bekeken Netflix-series die ooit uitgezonden is. In een krankzinnig sadistische ratrace – gebaseerd op ouderwetse kinderspelletjes – krijgen een paar honderd aan lager wal geraakte mensen de kans om nog wat van hun leven te maken. Hoewel de serie een leeftijdsgrens van 16 jaar en ouder kende, was de serie onder basisschoolleerlingen razend populair en werd massaal nagespeeld op het schoolplein, tot ongenoegen van ouders en leerkrachten. In het bloedstollende spel Glass Stepping Stones uit de zevende aflevering van de serie moeten 16 deelnemers een brug met 18 treden oversteken[1]. Eén voor één proberen de deelnemers veilig de overkant te bereiken. Bij elke trede heeft een deelnemer de keuze om het linkerpaneel of het rechterpaneel te kiezen, waarbij één van de twee panelen uit normaal glas bestaat dat meteen breekt als er op wordt gestapt en de andere paneel gehard glas bevat, waar veilig op gestapt kan worden zonder dat het glas breekt. Voor elke trede is er een kans van 50% dat het linkerpaneel gehard glas heeft, en als de linkerkant gehard glas heeft, heeft de rechterkant dat niet (en omgekeerd). Het is onmogelijk om het verschil te zien tussen het normale en het geharde glas. Het slechte nieuws is dat als een deelnemer op een paneel met normaal glas springt, het glas breekt en de deelnemer naar beneden tuimelt met de dood als gevolg. Het goede nieuws is dat het offer niet voor niets is geweest, want het gebroken paneel geeft alle overgebleven deelnemers waardevolle informatie over wat de juiste weg naar veiligheid is. Verder wordt verondersteld dat elke deelnemer ook de informatie heeft wat de veilige panelen zijn die door voorgaande deelnemers gekozen zijn. In volgorde probeert elke deelnemer de brug over te steken en de deelnemer blijft in beweging totdat de deelnemer ofwel succesvol alle 18 treden op de brug heeft overgestoken, ofwel tussentijds naar beneden is getuimeld. Wat is het verwachte aantal overlevende deelnemers, wat is voor elke deelnemer de kans op overleven en wat is de kansverdeling van het aantal deelnemers dat overleeft?

Dit spel is dodelijker dan het spel van Russisch roulette. Het Markovketen concept is ideaal voor een kanstheoretische analyse van het spel. Beschouw een absorberende Markovketen met 20 toestanden i = 0, 1, ..., 18, 19. Toestand i met $1 \leq i \leq 18$ betekent dat het spel gevorderd is tot trede i waar echter echter een deelnemer op het normale glas van deze trede gesprongen is en jammerlijk het leven gelaten heeft, toestand 19 betekent dat een deelnemer veilig op trede 18 beland is en dus de overkant bereikt heeft, en toestand 0 is een hulptoestand die het begin van het spel markeert. Toestand 19 wordt genomen als een absorberende toestand van de Markovketen, dat wil zeggen als het proces toestand 19 bereikt heeft dan blijft het daar in. Voor i = 0, 1, ..., 18, worden voor de Markovketen de één-staps overgangskansen p_{ij} van toestand i naar toestand j gegeven door

$$p_{ij} = \left(\frac{1}{2}\right)^{j-i} \text{ voor } j = i+1, .., 18 \quad \text{en} \quad p_{i,19} = \left(\frac{1}{2}\right)^{18-i}.$$

Voor toestand 19 is $p_{19,19} = 1$ en de overige p_{ij} zijn 0. Laat $\mathbf{P}$ de 20×20 matrix van één-staps overgangskansen zijn. De simpele berekeningen gaan nu als volgt. De matrix producten $\mathbf{P}^k$ worden berekend voor $k = 1, ..., 16$. Noteer met a_k de kans dat deelnemer k overleeft en met d_k de kans dat precies k deelnemers overleven voor $k = 0,1..., 16$. Dan

$$a_k = p_{0,19}^{(k)} \quad \text{voor } k = 0, 1, \ldots 16,$$

dus a_k wordt gegeven door het (0, 19)de element van $\mathbf{P}^k$. De d_j's kunnen vervolgens worden berekend met

$$d_{16-k+1} = a_k - a_{k-1} \quad \text{for } k = 1, 2, \ldots, 16,$$