



# Covid-19 Fake News Detectie

Een onderzoek over de relatie tussen metadata en nepnieuws vanuit een AI perspectief

15-06-2022

---

Hogeschool van Amsterdam

Auteurs: Alex Jongejans, Ali Ozcan, Bas Levering, Jesmo de Jong, Joerie Church

Opdrachtgever: Pascal Wiggers

Begeleider: Ed Kuijpers

## Managementsamenvatting

In de Covid-19 pandemie is op de Hogeschool van Amsterdam een HBO-ICT AAI minor project opgestart om fake nieuws te detecteren op twitter gerelateerd aan Covid-19. Na de brede focus van het eerste team en de focus op tweet-inhoud van team twee is de focus van dit onderzoek de invloed die metadata heeft op het classificeren van nepnieuws. De gedachte daarbij is dat metadata aan minder verandering onderhevig is dan tekstuele inhoud en dus een geschikte toevoeging zou kunnen zijn voor nepnieuws detectie over een langere termijn.

De dataset beschikt over een label die bestaat uit echte, neutrale en neppe tweets. Deze is in dit onderzoek opgesplitst in verschillende subsets waardoor er een dataset ontstond met alle metadata, een met alleen tekst-data en een met tekst metadata. Deze verschillende datasets zijn gebruikt om te onderzoeken welke typen metadata het best gebruikt kunnen worden.

Allereerst zijn verschillende machine learning modellen getest om een inschatting te krijgen van hoe goed metadata presteert op basis van de accuracy score. De meeste scores liggen daarbij tussen de 60 en 70%, afhankelijk van de gebruikte subset. De precision en recall scores per type tweet geven een andere inkijk in het gedrag van de machine learning modellen. Deze analyse laat zien dat het gedrag erg verschilt per type tweet. Dit komt waarschijnlijk voort uit de samenstelling van de dataset.

Vervolgens is gekeken naar hoe modellen presteren puur op tekst-data. Deze scores zijn lager dan de scores van de voorgaande teams, omdat er minder data beschikbaar was. Het beste tekst-model is een BERT model dat 81% accuracy scoort. Daarna zijn verschillende manieren gebruikt om tekst-data en metadata te combineren. Allereerst wordt een netwerk getest waarbij verschillende afzonderlijk voorgetrainde modellen worden samengevoegd. Dit is het concatenate netwerk. Vervolgens wordt het attention principe toegepast in een ander netwerk. In dit netwerk worden verschillende combinaties aan subsets als input gebruikt. De beste combinatie is tweet-inhoud en tekst metadata met een score van 75%.

De beste prestatie van metadata is een score van 74% door middel van een soft-voting ensemble model. De beste combinatie leidt tot een score van 79% in het concatenate model. er is gebleken dat het concatenate netwerk beter scoort, maar een stuk langer duurt om te trainen dan het attention netwerk of het ensemble netwerk.

De overkoepelende conclusie van het onderzoek is dat metadata tot zekere hoogte geschikt is om te gebruiken als data voor het detecteren van nepnieuws. De scores zijn lager dan wanneer alleen tekst gebruikt wordt, maar het verschil is beperkt. De metadata lijkt ervoor te zorgen dat de combinaties van tekst en metadata ook minder goed presteren dan alleen tekst. Echter zorgt het gebruik van metadata voor verhoogde houdbaarheid.

## Inhoud

<b>Managementsamenvatting</b>	<b>1</b>
<b>Inhoud</b>	<b>2</b>
<b>1. Inleiding</b>	<b>4</b>
1.1 Onderzoek	4
1.2 Leeswijzer	5
<b>2. Introductie tot metadata</b>	<b>5</b>
2.1 Soorten	6
2.1.2 Structureel	6
2.1.3 Administratief	6
2.1.4 Technisch	7
2.2 Twitter	7
<b>3. Data beschrijving</b>	<b>9</b>
3.1 Afkomst en bruikbaarheid	9
3.2 Inhoud	10
3.3 Vergelijkingsbasis	10
<b>4. Pre-processing</b>	<b>12</b>
4.1 Label	12
4.2 Objecten/lists	12
4.3 Ontbrekende waarden	13
<b>5 Feature Engineering</b>	<b>14</b>
5.1 Binaire variabelen vanuit NaN	14
5.2 Sentiment score	14
5.3 user_creation_tweet_diff	14
<b>6. Feature selectie</b>	<b>15</b>
6.1 Correlatiematrix	15
6.1.1 Correlatie verklarende variabelen met RFG	15
6.1.1 Onderlinge correlatie verklarende variabelen	16
6.2 T-toetsen	17
6.3 Chi Kwadraat toetsen	18
<b>7. Resultaten subdatasets zonder feature selection</b>	<b>19</b>
7.1 Subdatasets	19
7.2 Resultaten zonder feature selection	20
<b>8. Resultaten met feature selection</b>	<b>21</b>
<b>9. Feature en permutation importances</b>	<b>22</b>

9.1 Feature importance	22
9.2 Permutation Importance	23
<b>10. Recall en Precision</b>	<b>26</b>
10.1 Inleiding	26
10.2 Precision en Recall	26
10.3 SVM, KNN & RF Scores	27
10.4 Neurale Netwerk Scores	28
<b>11. Resultaten op tekstuele inhoud</b>	<b>29</b>
11.1 Modellen	29
11.2 Resultaten	30
<b>12. Combinatie tekst en metadata</b>	<b>31</b>
12.1 Voorgaande keuzes	31
12.2 Resultaten	32
12.2.1 Samenvatting	33
12.2.2 Vergelijking met de alleenstaande netwerken	33
<b>13. Ensemble Learning</b>	<b>34</b>
<b>14. Attention</b>	<b>36</b>
14.1 Wat is Attention?	36
14.2 Toepassen van Attention	36
14.3 Self Attention CyberZHG & Bi-LSTM	37
14.4 Attention layer	37
14.4.1 Resultaten	38
14.4.2 Vergelijking met Concatenate netwerk	39
<b>15. Explainable AI</b>	<b>40</b>
<b>16. Samenvattende plot</b>	<b>42</b>
<b>17. Conclusie</b>	<b>43</b>
<b>18. Aanbevelingen</b>	<b>46</b>
<b>19. Bibliografie</b>	<b>47</b>
<b>20 Bijlagen</b>	<b>49</b>
20.1 Bijlage A Concatenated neurale netwerk	49
20.2 Bijlage B Tabellen	50
20.3 Bijlage C Attention netwerk	53

## 1. Inleiding

"Twitter heeft maatregelen genomen tegen Nederlandstalige accounts die de afgelopen maanden nepnieuws over het coronavirus hebben verspreid" (RTL, 2020).

Hele websites en Twitter-accounts staan vol van berichten die soms totaal bij elkaar verzonnen zijn of bevatten fouten informatie. Bij het lezen van deze Tweets zou je je kunnen afvragen in hoeverre het waar is. Iedereen kan in een nepbericht trappen. "De gevolgen ervan kunnen groot zijn. Fake Berichten hebben zo de mogelijkheid om bijvoorbeeld verkiezingen te beïnvloeden" (AD, 2017). Aangezien deze heel snel verspreid kunnen worden, is het ook gemakkelijk om de publieke opinie te beïnvloeden.

Het is dus handig om nepnieuws te herkennen. Alleen vindt niet iedereen dit even makkelijk. Wat nou als je binnen één klik kunt weten of iets nep nieuws is? Dat is één van de dingen waar het Responsible AI Lab naar toe werkt. Het Responsible AI Lab werkt al een aantal jaar samen met Nederlandse media partijen zodat studenten aan AI-vraagstukken kunnen werken.

### 1.1 Onderzoek

Het herkennen van nepnieuws is nu al mogelijk, maar het kan altijd beter. De afgelopen twee jaar hebben projectgroepen van de minor Applied Artificial Intelligence (Applied AI) op basis van covid-19 specifieke Tweets geprobeerd om nepnieuws te detecteren. Deze projectgroepen zijn daarin geslaagd. Het beste AI-model heeft een accuracy van maar liefst 92% op de testdata gescoord. Hierbij is er alleen gebruik gemaakt van de inhoud van de tweets. Echter verandert de inhoud van Tweets snel. Covid-19 data is inmiddels al minder actueel. De voorgaande covid-19 tweet inhoud zijn daardoor minder waard.

Een tweet heeft veel meer eigenschappen dan alleen tekst, oftewel: metadata. Denk bijvoorbeeld aan de profielfoto of de gebruikersnaam van degene die de tweet heeft verzonden. Zelfs de locatie van het verzenden van de tweet hoort bij deze eigenschappen. Dit onderzoek richt zich op de relatie tussen metadata en nepnieuws. Hierdoor zouden modellen kunnen ontstaan die langer actueel blijven dan modellen gebaseerd op textuele inhoud.

De hoofdvraag voor dit onderzoek luidt als volgt: In hoeverre is nepnieuws op Twitter te detecteren aan de hand van Twitter metadata in vergelijking met detectie op grond van de tekstuele inhoud van een tweet?

Deze vraag wordt beantwoord met behulp van de volgende vragen:

- Wat is metadata?
- Welke modellen zijn geschikt voor het detecteren van nepnieuws?
- Welke features dragen het meest bij aan het detecteren van nepnieuws?

- Hoe presteert een model gefocust op tekstuele inhoud?
- Hoe presteert een model dat op alleen metadata getraind is?
- Op welke manieren kunnen modellen gebaseerd op tekst en/of metadata gecombineerd worden of elkaar ondersteunen met mogelijke toelichting van de resultaten?

## 1.2 Leeswijzer

In hoofdstuk 2 wordt de betekenis van metadata beschreven, welke soorten metadata er zijn en het soort dat wordt gebruikt voor dit onderzoek.

In de hoofdstukken 3 t/m 6 wordt het soort data beschreven en hoe de dataset opgeschoond is. Hoofdstuk 4 gaat over het pre-processen. Hier wordt besproken wat er gedaan wordt met lege kolommen en ongeldige waardes (zoals bijvoorbeeld niet direct bruikbare waardes). De stappen hoe de dataset bewerkt is worden hier uitgelegd. Hoofdstuk 5 gaat over de feature engineering. Door middel van bestaande variabelen zijn er namelijk nieuwe variabelen gemaakt. De feature selection wordt in hoofdstuk 6 behandeld.

Vanaf hoofdstuk 7 worden resultaten van de machine learning modellen en neurale netwerken besproken. In hoofdstuk 7 worden modellen getraind met data waar geen feature selection op is toegepast. Hoofdstuk 8 bespreekt modellen waar dit wel op is toegepast. Hoofdstuk 9 gaat in op de feature en permutation importance. Dit wordt gebruikt om te onderzoeken welke kolommen van de dataset belangrijk zijn. Hoofdstuk 10, recall en precision, bestudeert het gedrag van de gebruikte modellen, namelijk of ze systematisch dezelfde classificatie fouten maken.

Hoofdstuk 11 laat de resultaten van verschillende modellen zien die getraind zijn op de tekstuele inhoud van de tweets. In hoofdstuk 12 zijn de resultaten te vinden van het concatenate netwerk, een samenvoeging van verschillende modellen. Hoofdstuk 13 beschrijft de ensemble learning methoden die zijn toegepast op de machine learning modellen. In hoofdstuk 14 wordt het gebruikte attention netwerk beschreven evenals de resultaten ervan. Hoofdstuk 15 beschrijft een XAI methode die is toegepast op verschillende modellen om de bijdrage van verschillende variabelen te zien op de resultaten. In hoofdstuk 16 wordt een samenvattende plot weergegeven. In hoofdstuk 17 en 18 zijn de conclusies te vinden van het onderzoek en worden er aanbevelingen gedaan.

## 2. Introductie tot metadata

Sinds de opkomst van sociale netwerken en smartphones monitoren veel bedrijven zoals Facebook consumenten dagelijks. Persoonlijke data wordt opgeslagen, met of zonder jouw toestemming. Denk hierbij aan je naam en locatie, maar ook dingen zoals de posts die je op Instagram plaatst. Inmiddels is data een integraal deel van de maatschappij. Daarentegen is het woord metadata een stuk minder bekend. Wat houdt metadata dan precies in? Wat heb je aan metadata?

Metadata is kort gezegd: "data over data" (ICT Portal, 2018). Het zijn gegevens die data omschrijven. Metadata geeft de context die nodig is om data goed te kunnen beheren en begrijpen. Door metadata is het sneller duidelijk wat de betekenis is van de data, waar het vandaan komt sneller duidelijk en of het betrouwbaar is.

### 2.1 Soorten

Metadata wordt dus gebruikt voor de beschrijving van verschillende informatie. Er bestaan verschillende soorten metadata. Deze kunnen worden opgedeeld in vier categorieën.

#### 2.1.1 Beschrijvend

Beschrijvende metadata kunnen worden gezien als kenmerk. Denk aan bijvoorbeeld de titel van een webpagina, door wie iets is gemaakt, de taal waarin iets is geschreven en wanneer een website opgesteld is.

#### 2.1.2 Structureel

Structurele metadata beschrijft hoe de componenten van een object zijn georganiseerd. Een voorbeeld hiervan is hoe pagina's worden geordend om hoofdstukken van een blog te vormen. Ook worden structurele metadata gebruikt om de zoekresultaten op het internet of in een intern systeem te beïnvloeden (ICT Portal, 2018). Er zijn bijvoorbeeld metadata die voor een verbeterd Search Engine Optimization kunnen zorgen.

#### 2.1.3 Administratief

Administratieve metadata zorgen ervoor dat elementen gecategoriseerd en geordend worden (ICT Portal, 2022). Het helpt om de bron te beheren. Dit is vooral van toepassing in een systeem voor documentbeheer. Een voorbeeld van administratieve metadata is het bewaren van documenten.

### 2.1.4 Technisch

Technische metadata beschrijven de technische eigenschappen van een element zoals de bestandsgrootte, de verschillende bewerkdata en type bestand (ICT Portal, 2018) Deze metadata worden vaak gebruikt om de communicatie tussen systemen te vergemakkelijken. Een webservice kan bijvoorbeeld op basis van de technische metadata van een afbeelding beoordelen of het element niet te groot is om te uploaden in het systeem.

## 2.2 Twitter

Dit verslag focust zich op het gebruiken van de metadata van tweets. Echter is Twitter slechts een voorbeeld van een online platform waarbij metadata een rol speelt. Platformen zoals Facebook, Reddit en Twitter stellen gebruikers in staat om onder andere weblinks, documenten, video's, meningen en informatie met elkaar te delen. Bij het gebruik van deze platformen komt metadata vrij. Tijdens het plaatsen van een tweet worden bijvoorbeeld 144 verschillende stukken metadata gegenereerd naast de 140 karakters in de tweet zelf (Perez et al., 2018).

Beatrice Perez van University College London, co-auteur van het onderzoeksrapport *You are your metadata: Identification and Obfuscation of Social Media Users using Metadata Information*, beschrijft in een interview de attitude van mensen richting metadata: "Mensen nemen onterecht aan dat ze niet geïdentificeerd kunnen worden, omdat hun data online staat" (B. Greif, 2018). Daarentegen zou niemand bijvoorbeeld hun woonadres delen met een compleet vreemd persoon. Mensen zijn zich er simpelweg niet van bewust dat hun metadata gebruikt kan worden om hun te identificeren. (Greif, 2018).

Metadata van sociale media wordt actief gebruikt. Deze metadata wordt zelfs gebruikt op manieren die niet overeenkomen met de originele doelen van de platformen. Informatie die verzameld wordt voor advertentie doeleinden kan bijvoorbeeld worden ingezet om politieke en religieuze achtergronden van een gebruiker te achterhalen (Perez et al., 2018). Zo kan er bijvoorbeeld door tekstanalyse het geslacht, de leeftijd, de politieke oriëntatie of gemoedstoestand van een gebruiker te achterhalen.

Ondanks de onwetendheid van consumenten op het gebied van metadata zijn er ook voorbeelden van metadata die wel niet onbekend zijn bij gebruikers. In het specifieke geval van Twitter zijn bijvoorbeeld het aantal volgers, retweets of favorites bekende maten. Metadata is inmiddels een kern component geworden van de services die sociale media platformen bieden. De hiervoor genoemde voorbeelden zijn namelijk niet slechts extra informatie. Twitter gebruikers vertrouwen op dit soort maten om de authenticiteit van een account te beoordelen (Perez, 2018).





Het doel van het onderzoek van Perez is om het identificatie risico van een Twitteraccount puur gebaseerd op metadata te illustreren. Door machine learning modellen te trainen op basis van tweets van ruim 5 miljoen gebruikers kan een enkele gebruiker met 96.7% accuraatheid geïdentificeerd worden uit een groep van 10.000 (Greif, 2018,). Deze uitkomst illustreert de kracht van het gebruik van metadata.

## 3. Data beschrijving

### 3.1 Afkomst en bruikbaarheid

Om de uitkomsten van dit project te kunnen vergelijken met de behaalde scores uit de twee voorgaande onderzoeken, wordt er gebruik gemaakt van dezelfde dataset. Er is wel onderzoek gedaan naar de mogelijkheden om alternatieve datasets in het onderzoek op te nemen. Echter, er is snel de conclusie getrokken dat bruikbare twitter datasets schaars zijn. Verder zou de vergelijkingsbasis met de vorige twee projecten een stuk ingewikkelder gemaakt worden. Dit zijn de twee voornaamste redenen waarom dezelfde dataset gebruikt is als in de vorige twee projecten van het Responsible AI lab.

Een overige reden voor het gebruik van de dataset is dat de informatie in de dataset op een andere wijze gebruik wordt. In de voorgaande projecten is er nadrukkelijk gebruik gemaakt van de tekst inhoud van tweets. Echter wordt in dit project de focus gelegd op de metadata in de dataset. Ondanks het gebruik van dezelfde dataset, wordt er dus gebruik gemaakt van verschillende aspecten en informatie uit de dataset. Het werken met dezelfde dataset heeft ook als gevolg dat de modellen uit voorgaande onderzoeken gecombineerd kunnen worden met de modellen die in dit project tot stand komen.

De dataset is tot stand gekomen door twee verschillende datasets samen te voegen, namelijk de Constraint 2021 dataset en de CMU MisCov-19 dataset. Constraint 2021 is een competitie dat actief begon in oktober 2020. Verschillende professoren van universiteiten in verschillende landen hebben de dataset tot stand gebracht. De CMU MisCov-19 dataset is afkomstig van onderzoekers van de Carnegie Mellon University in Pittsburgh, Pennsylvania (Flietstra et al., 2021).

De betrouwbaarheid van de dataset wordt onderbouwd met het argument dat de makers van beide datasets zijn afkomstig van gerenommeerde instanties. De auteurs van de Constraint 2021 dataset zijn namelijk allen data wetenschappers aan de Indian Institute of Information Technology Sri City en de makers van de CMU MisCov 19 zijn afkomstig van de Carnegie Mellon University.

Het opnieuw gebruiken van de dataset heeft een aantal gevolgen die belangrijk zijn om op te noemen. De labels van de dataset kunnen inmiddels achterhaald zijn, aangezien een tweet gelabeld als fake informatie kan bevatten die tegenwoordig wel aantoonbaar waar is. Een ander nadeel van het gebruik van de dataset is het feit dat de tweets verzameld zijn door mensen en er daardoor een bias in kan zitten.

## 3.2 Inhoud

De dataset bestaat uit 98 variabelen afkomstig uit de Application Programming Interface (API) die verschillende aspecten van de bijna 8000 tweets beschrijven. De variabelen kunnen in groepen worden ingedeeld.

Naast de tekstinhoud van de tweet zelf zijn er twee variabelen die in de vorm van tekst staan: de username en de description (biografie). Daarnaast zijn er een aantal variabelen die informatie verschaffen over de inhoud van de tweet, zoals het aantal gebruikte hashtags, urls en de lengte van de tweet. De dataset verschaft ook informatie over de locatie en het tijdstip waarop de tweet verzonden is en over de activiteit rondom de tweet zoals het aantal retweets of de favourite count. Verder bevat de dataset features die het profiel van de gebruiker in kaart brengen, zoals of de gebruiker het standaard format voor het profiel gebruikt.


## 3.3 Vergelijkingsbasis

Om de resultaten van dit onderzoek te vergelijken met het vorige onderzoek is het belangrijk om een zo goed mogelijke vergelijkingsbasis te creëren. Een voorwaarde voor het vergelijken van de resultaten is dat dezelfde dataset gebruikt wordt.

De CMU MisCov-19 dataset bevat metadata en de tekst van de tweets. Echter bevat de Constraint 2021 dataset naast de real fake grade alleen de tekst van een tweet. Team 1 heeft de datasets samengevoegd een script gebruikt om de metadata op te halen van de tweets uit de Constraint 2021 dataset. Echter waren ze niet in staat om alle metadata op te halen, omdat er inmiddels tweets van twitter verwijderd worden. De reden hiervoor is dat twitter zelf fake news verwijderd. In de gecombineerde dataset die de basis vormt voor de onderzoeken van team 1 en team 2 zijn er dus nog steeds tweets waarvan alleen de tekst bekend is. In de onderstaande figuur wordt dit verder uitgelegd. Om op een eenvoudige wijze te laten zien hoe team 1 en team 2 de data gebruikt hebben is de data in de figuur opgesplitst in 4 kwanten, waarvan het kwart links onderin leeg (zwarte vak) is, omdat de metadata van die tweets niet opgehaald kon worden.

Data Team 1			Data Team 2		
Rij	Metadata	Tekstdata	Metadata	Tekstdata	
7907 →					
13565 →					

figuur 1 : Schematisch overzicht van het gebruik van de data door teams 1 en 2.



Team 1 heeft de eerste 7907 tweets uit de dataset gebruikt. Team 2 heeft zich echter volledig gefocust op tekst data, waardoor zij alle 13565 rijen aan data gebruikt hebben. Voor het trainen van onze modellen worden de eerste 7909 rijen gebruikt, aangezien dit onderzoek zich richt op de combinatie tussen metadata en tekst.

De resultaten van dit onderzoek kunnen dus niet eenvoudig vergeleken worden met de resultaten die team 2 behaald heeft, omdat zij meer trainings data tot hun beschikking hadden.

## 4. Pre-processing

De oorspronkelijke dataset bestaat uit 7909 rijen en 98 kolommen. Echter is deze dataset nog niet geschikt voor AI modellen. De dataset bevat bijvoorbeeld veel lege cellen en NaN-waardes. Bovendien zijn er kolommen die objecten of één unieke waarde bevatten. Ook hier kunnen modellen weinig mee. De dataset moet dus bewerkt worden op een zodanig manier dat er iets mee gedaan kan worden. In dit hoofdstuk wordt uitgelegd hoe de oorspronkelijke dataset stapsgewijs is bewerkt.

### 4.1 Label

De target variabele uit de dataset is de variabele “real\_fake\_grade” (RFG). De type van dit kolom is een float en bevat tien unieke waardes (inclusief NaN). Het eerste wat gedaan is, is het verwijderen van tweets waarvan het onbekend is of een tweet fake, real of neutraal is (NaNs). Na het verwijderen van de NaN cellen, zijn er nu nog 7907 rijen over.

Daarnaast zijn de 9 overgebleven unieke scores veranderd naar drie verschillende waardes: 1 (echt), 0 (neutraal) en -1 (nep). Het voorgaande project (Team 2) heeft ook gebruik gemaakt van deze scores.

De RFG uit de oorspronkelijke dataset werkt als een schaal waar een neutrale waarde tweet voornamelijk betrekking heeft tot politieke, satirische tweets of tweets die over covid gaan, maar hier niks over zeggen.. Een tweet die over covid gaat, maar hier niks over zegt is bijvoorbeeld: “Ik heb covid.” Als de RFG score van een tweet meer naar 1 (real) oploopt, dan bevat de tweet meer feiten of nuttige informatie. Wanneer de RFG score daalt naar -1 (fake), dan bevat de tweet meer nepnieuws of een vorm van complottheorie.

### 4.2 Objecten/lists

De oorspronkelijke dataset bevat kolommen met objecten. Een object is een dictionary met key-value pairs, waarbij de key een variabele is en de value de bijbehorende waarde. Objecten zijn dus eigenlijk verzamelingen van meerdere variabelen. De objecten zijn uitgesplitst zodat iedere variabele die oorspronkelijk in een object stond individueel gebruikt kan worden. Zo was ‘hashtags\_count’ eerst een onderdeel van ‘entities’. In de nieuwe dataset heeft ‘hashtags\_count’ zijn eigen kolom gekregen.

### 4.3 Ontbrekende waarden

Sommige kolommen zijn verwijderd, omdat ze veel waarden missen. Een aantal variabelen had nuttig kunnen zijn wanneer ze genoeg data hadden bevat. Er is gekozen om variabelen te verwijderen wanneer de variabele uit 75% of meer aan NaN waarden bestaat. Echter is het niet altijd zo dat de NaN waarden informatieloos zijn. Sommige variabelen, bijvoorbeeld of de locatie van een gebruiker aan staat, bevatte een heel groot aantal NaN waarden. De NaN waarde is in dat geval een indicatie dat de gebruiker geen locatiegegevens deelt. Dit soort informatie is gebruikt om binaire variabelen mee te maken. Hierover wordt in het volgende hoofdstuk meer verteld.

## 5 Feature Engineering

in de laatste paragraaf van het vorige hoofdstuk is aangestipt dat veel variabelen een grote hoeveelheid NaN waarden bevatten. in dit hoofdstuk worden onder andere deze features besproken. Er wordt een duidelijker beeld geschetst van hoe er met deze variabelen omgegaan is om zo veel mogelijk bruikbare data uit de dataset te halen. Daarnaast worden overige zelf gecreëerde variabelen besproken.

### 5.1 Binaire variabelen vanuit NaN

In paragraaf 4.3 is een voorbeeld gegeven van een variabele die omgezet is naar een binaire variabele wegens een grote hoeveelheid NaN waarden. De dataset bleek een groot aantal variabelen te bevatten die op deze manier toch bruikbaar zijn. Voorbeelden hiervan zijn of een tweet een url bevat, of een twitter gebruiker Apple of Android gebruikt en of een gebruiker geverifieerd is. Voor alle binaire variabelen kan de file preprocessing\_2022.ipynb geraadpleegd worden. Hierin worden onder andere deze variabelen aangemaakt.

### 5.2 Sentiment score

Er is gekozen een feature toe te voegen die beschrijft wat het sentiment is van een stuk tekst: de sentiment score. Deze is geïmplementeerd met een pretraind BERT model. Dit model geeft een score terug gebaseerd op de ingevoerde tekst die aangeeft hoe positief of negatief het sentiment in de tekst is. Om dit te bewerkstelligen is de tekst omgezet in tokens (losse woorden) zodat deze kunnen worden gebruikt als input sequence.

De sentimentanalyse kan alleen gedaan worden op tekst data. in de dataset bestaan drie kolommen die in aanmerking komen voor een sentiment score: username (gebruikersnaam), description (beschrijving van een gebruiker) en natuurlijk de tweet tekst zelf.

### 5.3 user\_creation\_tweet\_diff

Er is een feature gemaakt die kijkt naar de leeftijd van een account bij het maken van de tweet. De reden hierachter is dat fake news wellicht wordt verspreid door bots en botnets. Het zou kunnen dat een groot aantal bots tweets meteen plaatst na het aanmaken van een account. In andere woorden betekent dit dat het verschil tussen het maken van een account en het versturen van de tweet laag is. Uit de Twitter API komen twee tijden die in de dataset (als string) in een dag-maand-jaar-tijdstip format staan. Dit zijn het tijdstip waarop de account aangemaakt is en het tijdstip waarop de tweet verzonden is. Deze tijden worden gerepresenteerd door de variabelen user\_created\_at en created\_at. Deze zijn omgezet naar epoch tijd (dit is het aantal seconden na 1 Januari 1970). Vervolgens worden de twee tijden van elkaar afgetrokken om de variabele user\_creation\_tweet\_diff te maken.

## 6. Feature selectie

In het vorige hoofdstuk is beschreven hoe de dataset is opgeschoond. Na dit proces is gekeken naar de relevantie van de overgebleven kolommen. Dit hoofdstuk beschrijft de stappen die genomen zijn om te bepalen of een kolom bedraagt aan het model.

### 6.1 Correlatiematrix

#### 6.1.1 Correlatie verklarende variabelen met RFG

In eerste instantie is gekeken naar de onderlinge correlatie tussen variabelen door middel van een correlatiematrix. De focus bij het beoordelen van de correlaties ligt op de correlatie tussen een bepaalde variabele en de responsvariabele RFG. Verder geeft de matrix ook een beeld van de verklarende variabelen die onderling sterk correleren en daardoor mogelijk de prestaties van de modellen kunnen verminderen.

Opvallend is dat geen enkele variabele sterk correleert met de RFG. Vrijwel alle correlatiecoëfficiënten van verklarende variabelen met de RFG zijn kleiner dan 0.1. De verklarende variabele met de sterkste correlatie met de *real\_fake\_grade* is *user\_is\_verified*, namelijk 0.33. Een verified user is een gebruiker die een blauw vinkje achter zijn profiel heeft gekregen om aan te tonen dat het daadwerkelijk om de echte persoon gaat en niet om bijvoorbeeld een fan account.

De tweede sterkst correlerende variabele is *part\_of\_thread* met een score van 0.26. Een thread is een serie van aaneengeschakelde tweets van een enkele gebruiker. Dit stelt een gebruiker in staat om een verhaal te vertellen of informatie te delen over meerdere tweets in plaats van beperkt te blijven tot het aantal karakters in een enkele tweet.

Verder is de correlatiecoëfficiënt van *user\_service\_level\_media\_studio* 0.22. Deze variabele geeft aan of de account geregistreerd is als media studio. Een voorbeeld van een ander service level is een adverteerder.

Al met al zijn er geen sterke correlaties te vinden met de RFG. Daarentegen is dit geen nutteloze uitkomst. Het verschil tussen de correlatiecoëfficiënten van de drie hierboven genoemde variabelen en veel overige variabelen (met een score van bijna nul) kan een indicatie zijn dat deze drie variabelen in ieder geval een redelijke invloed uitoefenen op voorspellingen ten opzichte van de nihil invloed van overige variabelen.



### 6.1.1 Onderlinge correlatie verklarende variabelen

Naast de correlatie tussen de verklarende variabelen en de RFG spelen de onderlinge correlaties tussen de verklarende variabelen ook een rol. Deze kunnen namelijk voor ruis zorgen en daardoor leiden tot slechtere prestaties van machine learning algoritmen. Hieronder volgt een overzicht van alle variabelen die zeer sterk onderling gecorreleerd zijn. Een correlatie wordt als zeer sterk bestempeld wanneer de correlatiecoëfficiënt groter is dan 0.8 of kleiner dan -0.8.

Variabele 1	Variabele 2	Correlatiecoëfficiënt
<i>user_followers_count</i>	<i>user_normal_followers_count</i>	1
<i>user_listed_count</i>	<i>user_followers_count</i> ( <i>user_normal_followers_count</i> )	0.94 (0.94)
<i>retweet_count</i>	<i>favorite_count</i>	0.89
<i>user_service_level_mms</i>	<i>user_has_translation_enabled</i>	0.81

Tabel 1: Overzicht van zeer sterk gecorreleerde verklarende variabelen uit de dataset.

Variabelen die niet zeer sterk gecorreleerd zijn, maar nog steeds een sterk verband vertonen zouden ook ruis kunnen introduceren. De tabel hieronder bevat sterk gecorreleerde onderlinge variabelen: een correlatiecoëfficiënt tussen de 0.6 en 0.8 of -0.6 en -0.8.

Variabele 1	Variabele 2	Correlatiecoëfficiënt
<i>possibly_sensitive_media</i>	<i>tweet_contains_url</i>	-0.72
<i>user_listed_count</i>	<i>user_service_level_media_studio</i>	0.71
<i>user_service_level_media_studio</i>	<i>user_service_level_dso</i>	0.7
<i>user_has_default_profile</i>	<i>user_creation_tweet_diff</i>	-0.68
<i>user_service_level_media_studio</i>	<i>user_follower_count</i> ( <i>user_normal_follower_count</i> )	0.64 (0.64)

Tabel 2: Overzicht van sterk gecorreleerde verklarende variabelen uit de dataset.

Tabel 2 toont een aantal interessante verbanden, maar ook een aantal correlaties die op basis van domeinkennis minder snel te verklaren zijn. De correlatie tussen *possibly\_sensitive\_media* en *tweet\_contains\_url* bedraagt -0.72. Een negatieve correlatiecoëfficiënt is een indicatie dat de twee variabelen in tegenovergestelde richtingen bewegen. Dit betekent dat een tweet minder urls bevat wanneer de kans op gevoelige inhoud groter wordt.

Verder blijkt dat *user\_has\_default\_profile* en *user\_creation\_tweet\_diff* ook onderling negatief correleren. Dit betekent dat de kans op een standaardprofiel toeneemt, wanneer de het tijdsverschil tussen het ontstaan van het account en het plaatsen van de tweet afneemt. In de praktijk is dit een verband dat logisch lijkt te zijn. Wanneer iemand net een account heeft aangemaakt en meteen een tweet wilt versturen, dan is de kans kleiner dat hij tijd heeft genomen op zijn profiel instellingen aan te passen dan wanneer hij een dag later pas zijn eerste tweet verstuurd. Het gedrag van bots zou ook tot uiting kunnen komen in dit verband, aangezien bots snel na het aanmaken van een account een tweet kunnen versturen. De variabele *user\_creation\_tweet\_diff* lijkt dus een variabele te zijn die wel degelijk invloed zou kunnen uitoefenen bij het herkennen van bots en nep tweets.

Op basis van de bovenstaande correlaties zou het verwijderen van een van aantal van de variabelen kunnen leiden tot minder ruis en betere resultaten. Dit wordt in een volgend hoofdstuk getoetst.

## 6.2 T-toetsen

De dataset bevat in totaal 39 binaire variabelen. Met Welch's is onderzocht of voor de twee categorieën in de binaire variabele, bijvoorbeeld tweets met een url en zonder een url, de gemiddelde RFG verschilt. Welch's t-toetsen compenseren voor ongelijke varianties mocht dit het geval zijn. De nulhypothese van de t-toetsen is dat er voor de twee categorieën van de binaire variabele geen verschil in het gemiddelde van de RGF zit. De alternatieve hypothese luidt dat er wel een significant verschil bestaat. Het significantieniveau is 0.05. De tabel op de volgende bladzijde geeft de p-waarden voor alle variabelen waarvan de p-waarde groter is dan 0.05. Voor deze variabelen is er niet voldoende aanleiding gevonden om de nulhypothese te verwerpen.

Achteraf is gebleken dat de Welch's t-toets niet de correcte toets is om te gebruiken vanwege het feit dat er geen (bij benadering) normaal verdeelde variabelen getoetst worden, maar binaire variabelen ten opzichte van een categorische variabele (de RFG). Het is daarentegen wel interessant dat de uitkomsten van de Welch's t-toets indirect geleid hebben tot hogere accuracy scores door het verwijderen van enkele variabelen (zie hoofdstuk 8), waardoor ervoor gekozen is om met deze resultaten door te gaan.

Variabele	p-waarde
<i>user_created_in_daypart_night</i>	0.79
<i>user_profile_location</i>	0.74
<i>twitter_apple_user</i>	0.48
<i>user_created_in_daypart_morning</i>	0.42
<i>user_translator_type</i>	0.41
<i>is_quoted_tweet</i>	0.26
<i>user_service_level_subscription</i>	0.23

Tabel 3: Overzicht van binaire variabelen met een p-waarde  $> 0.05$  als uitkomst van de Welch's t-toets.

Wederom kunnen de resultaten gebruikt worden om variabelen te verwijderen uit de dataset om daarmee te resultaten van de modellen te verbeteren.

### 6.3 Chi Kwadraat toetsen

Met de  $\chi^2$  toets is er gekeken hoe onafhankelijk de kolommen van de dataset zijn vergeleken met de RFG. De toets heeft geen normale verdeling nodig in een van de twee vergeleken kolommen. De test gaat na of de waargenomen waarde afwijkt van de verwachte (of gemiddelde) waarde en berekent daarmee het totaal van de gewogen kwadratische afwijkingen tussen deze waarden. Deze score wordt gebruikt om categorische variabelen in een classificatietaak te evalueren.  $\chi^2$  geeft twee waarden terug voor elke kolom, de p-waarde en de  $\chi^2$  score. Hoe hoger de  $\chi^2$  score hoe sterker de relatie van de kolom met de RFG. Een kolom met een zwakke relatie met de RFG draagt minder bij aan de voorspellingen van het model. De  $\chi^2$  toets is gebruikt om te bekijken of er een aantal variabelen uit de dataset verwijderd kunnen worden. De toets heeft geen nieuwe resultaten opgeleverd, maar het heeft wel tot een bevestiging geleid van de uitkomsten van de t-toets. Een aantal variabelen zou namelijk ook op basis van de  $\chi^2$  toets verwijderd worden.

## 7. Resultaten subdatasets zonder feature selection

### 7.1 Subdatasets

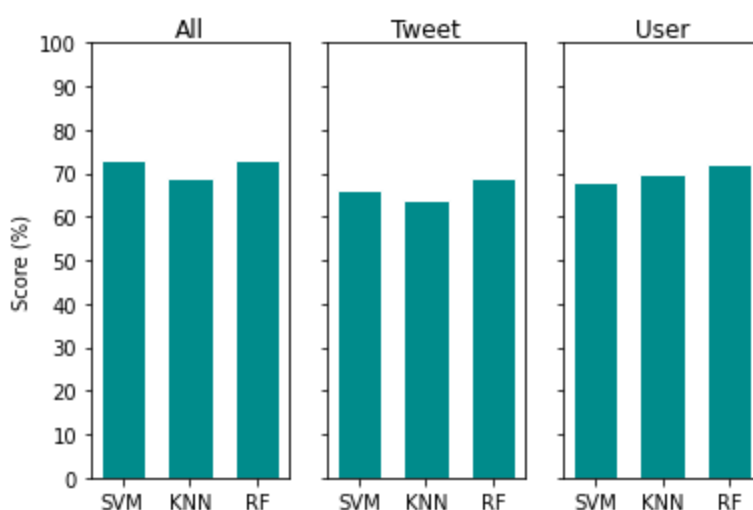
Om te kunnen onderzoeken welke soort metadata een sterke indicator kan zijn van nepnieuws (of echt nieuws) wordt de opgeschoonde dataset onderverdeeld in 3 subdatasets. De eerste subset bevat enkel kolommen die de tweets beschrijven, zoals het aantal retweets, hashtags etc. De tweede subset beschrijft de account door middel van variabelen zoals het aantal volgers, het type account (service level) en of het account de standaard profiel instellingen gebruikt. De derde subset bestaat uit 3 kolommen die tekst variabelen bevatten: de inhoud van de tweet zelf, de gebruikersnaam of de biografie van het account.

In totaal worden er dus 5 datasets gebruikt: de opgeschoonde (gehele) dataset, de opgeschoonde dataset waar feature selection op is toegepast en de 3 subdatasets: tweet data, user data en tekst data.

Om onafhankelijk van een machine learning algoritme een conclusie te kunnen trekken over de mate waarin bepaalde metadata een indicatie is van nepnieuws worden verschillende algoritmes getest op de verschillende datasets. Door bijvoorbeeld alleen rekening te houden met de resultaten van het best presterende algoritme zou er een conclusie getrokken kunnen worden die niet algemeen interpreteerbaar is, omdat er dan een bepaalde afhankelijkheid van een modelsoort kan bestaan. De resultaten van de verschillende algoritmen op de subsets aan metadata worden in de volgende paragraaf besproken. De resultaten op tekst-data worden in een apart hoofdstuk (hoofdstuk 12) behandeld omdat tekst data andere modelsoorten vereist dan de metadata.

## 7.2 Resultaten zonder feature selectie

De onderstaande figuur geeft de accuracy scores weer op de verschillende subsets aan metadata: alle metadata, tweet metadata en de user metadata. Middels een randomsearch is geprobeerd om de resultaten tot een bepaalde hoogte te optimaliseren. Er is echter daarna geen verdere optimalisatie uitgevoerd wegens de tijdsindeling van het project. Het doel is ook niet om de scores van deze algoritmen zo hoog mogelijk te krijgen, maar om te kijken of er patronen te herkennen zijn en deze te vergelijken met andere resultaten uit het project, zoals de resultaten met feature selectie die in de volgende hoofdstuk aan bod komen.



Figuur 2: Accuracy scores per subset van SVM, KNN, en RF machine learning modellen.

De bovenstaande figuur laat zien dat de scores van de modellen onderling ongeveer maximaal van 5% van elkaar verschillen. Dit geldt voor de resultaten op alle drie de subsets. De tweet metadata scoort gemiddeld lager dan de andere twee datasets. Voor de exacte percentages zie tabel 1 in bijlage B Tabellen.

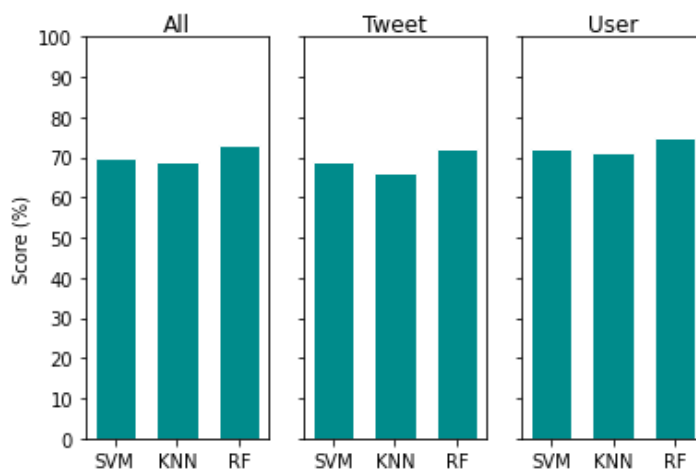
## 8. Resultaten met feature selection

Het vorige hoofdstuk beschrijft de resultaten waarbij geen vorm van feature selectie is toegepast. Alle variabelen uit de oorspronkelijke dataset zijn daarin meegenomen. In dit hoofdstuk worden de resultaten besproken die voortkomen uit dezelfde methodiek, maar op datasets waaruit een aantal variabelen verwijderd zijn.

De opgeschoonde dataset bevat 55 verklarende variabelen. Hieruit zijn 11 variabelen verwijderd. Van alle paren verklarende variabelen waarbij de onderlinge correlatie coëfficiënt groter is dan 0.8 is een variabele uit de dataset verwijderd. Deze variabelen zijn: *user\_normal\_followers\_count*, *user\_listed\_count*, *favorite\_count* en *user\_service\_level\_mms*. De variabelen die op basis van de Welch's t-toets en Chi-kwadraat toets zijn verwijderd zijn: *user\_created\_in\_daypart\_night*, *user\_profile\_location*, *twitter\_apple\_user*, *user\_created\_in\_daypart\_morning*, *user\_translator\_type*, *is\_quoted\_tweet*, en *user\_service\_level\_subscription*.

In totaal zijn er 3 variabelen uit de tweet data subset verwijderd, namelijk *favorite\_count*, *twitter\_apple\_user* en *is\_quoted\_tweet*. De overigen zijn ook verwijderd uit de user data subset. De onderstaande resultaten zijn verkregen op dezelfde wijze als in het vorige hoofdstuk om de resultaten zo goed mogelijk met elkaar te kunnen vergelijken.

De onderstaande figuur is bijna identiek aan die van resultaten van het vorige hoofdstuk. De verschillen zijn echter te klein om in deze plots te zien. Echter zijn alle scores van de tweet en user data met zo'n 1 tot 4% verbeterd. De exacte percentages zijn te vinden in tabel 2 van bijlage B. Aangezien de scores iets verbeterd zijn ten opzichte van de dataset zonder feature selectie en het aantal variabelen verkleind is, wat het model versimpeld is ervoor gekozen om in de hierop volgende hoofdstukken verder te gaan met de data waar feature selectie op is toegepast.



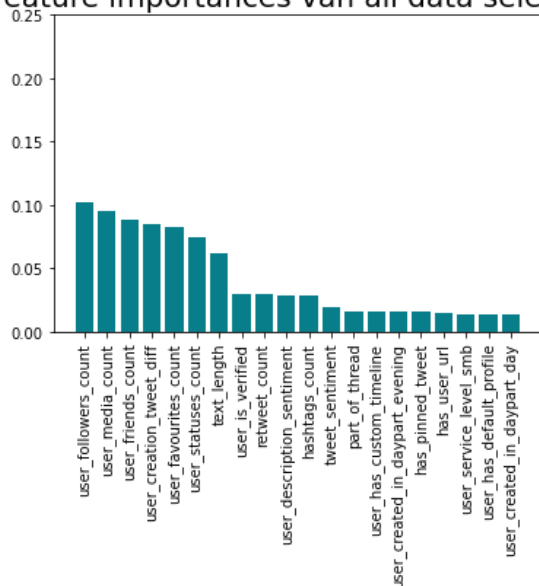
Figuur 3: Accuracy scores per modelsoort en dataset.

## 9. Feature en permutation importances

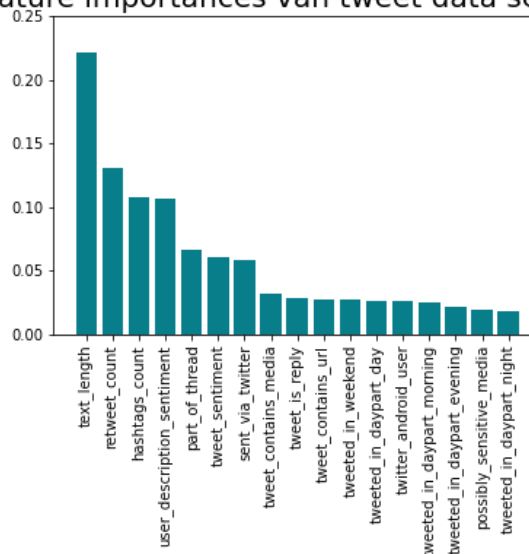
### 9.1 Feature importance

Deze paragraaf besteed aandacht aan de invloed van de individuele variabelen op de uitkomsten van het random forest model. Hieronder staan drie figuren met de 20 variabelen met de hoogste feature importance uit de betreffende dataset. Deze horen bij de datasets waar feature selection op is toegepast. In de titels van de figuren wordt hiernaar gerefereerd door de term 'selected'.

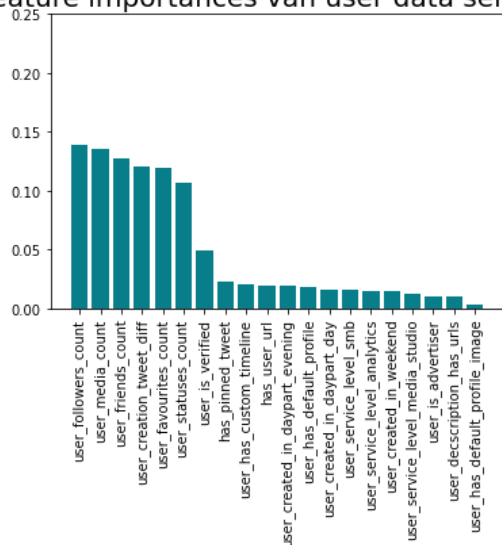
Feature importances van all data selected



Feature importances van tweet data selected



Feature importances van user data selected



Figuur 4: Feature importances per dataset. Nummering A (all data), B (tweet data), C (user data)

Uit de grafieken blijkt dat er voor elke dataset een aantal variabelen zijn die duidelijk meer invloed hebben dan anderen. Voor de gehele 'selected' dataset geldt dat de 'count' variabelen zoals het aantal volgers, vrienden en favorites zwaar meewegen. Daarnaast spelen de activiteit van een gebruiker (user\_statuses\_count) en hoe vaak er gebruik wordt gemaakt van een afbeelding of video (user\_media\_count) ook een invloedrijke rol ten opzichte van de andere variabelen.



Uit de feature importances van de tweet data blijkt dat de tekst lengte voor dit specifieke model wederom 'count' variabelen zoals het aantal retweets of de lengte van de tekst veel waarde toekent, maar dat de sentiment score van de biografie van een account en of een tweet deel uitmaakt van een thread ook relatief invloedrijk zijn.

Uit de user data blijkt dat de `user_creation_tweet_diff` (het verschil tussen het tijdstip van het plaatsen van de tweet en de creatie van de account) ook een relatief hoge feature importance heeft.

## 9.2 Permutation Importance

Opvallend is dat de variabelen die op basis van de bovenstaande figuren als meest invloedrijk bestempeld worden allemaal variabelen zijn die iets tellen (bijv. het aantal retweets of hashtags) terwijl de dataset een groot aantal binaire variabelen bevat. Of een gebruiker geverifieerd is en of de tweet deel uitmaakt van een thread zijn voorbeelden van binaire variabelen die op basis van de feature importance wel een relatief grote invloed lijken uit te oefenen.

Uit de documentatie van sklearn blijkt dat de feature importances een zeer hoge bias hebben richting variabelen die veel verschillende waarden aan kunnen nemen. Hierdoor kan er onterecht belangrijkheid toegekend worden aan numerieke variabelen t.ov. binaire variabelen. De numerieke variabelen in onze dataset zijn variabelen die een telling bijhouden. De figuren uit de vorige paragraaf zijn op zichzelf staand dus niet veelzeggend. Sklearn geeft als alternatief voor de feature importance de permutation importance.

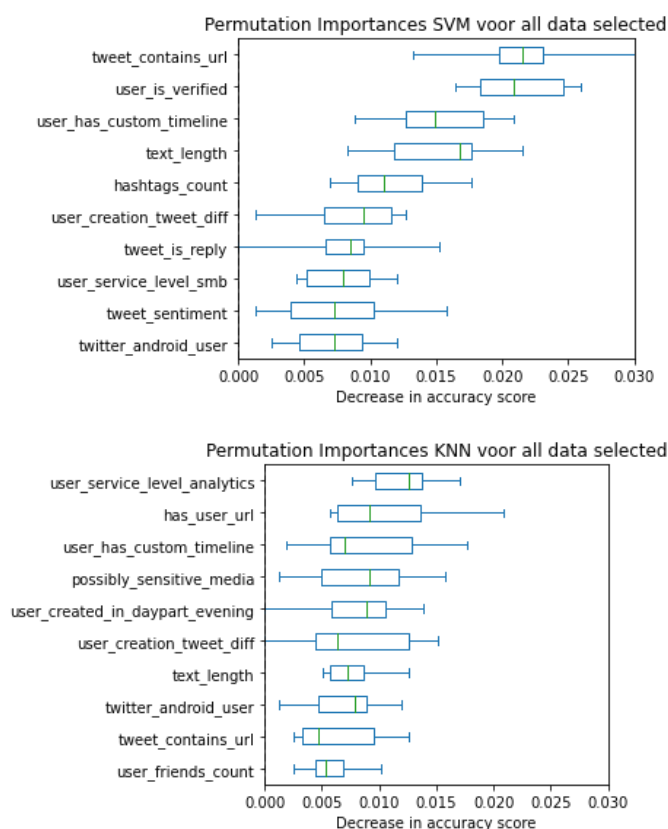
Hieronder volgt een uitleg over wat de permutation importance precies is. Daarna worde

Permutation importance wordt gedefinieerd als de afname van een prestatiemaat ten gevolge van het willekeurig onderling verschuiven van de waarden van deze feature. In ons geval wordt de accuracy gebruikt als prestatiemaat. Een feature wordt als belangrijk beschouwd wanneer de accuracy afneemt als gevolg van het verschuiven van de waarden van de feature. Dit betekent namelijk dat het model zijn voorspelling sterk baseert op de deze waarden (Molner, 2022). Stel dat de accuracy onveranderd blijft na het schuiven van de waarden van feature A, dan draagt A dus niet bij aan de voorspellende kracht van het model. De permutation importance verbreekt de relatie tussen de feature en de target, waardoor een afname in de prestatiemaat dus een indicatie is van hoe erg het model afhangt van de feature. Belangrijk om op te merken is dat net als bij feature importance, de permutation importance geen weergave is van de intrinsieke voorspellende kracht van een variabele, maar een maatstaf is van hoe belangrijk de feature is voor het specifieke model dat behandeld wordt.



De permutation importance plots geven voor elke variabele een boxplot die hoort bij de daling in accuracy score van het verschuiven van waarden van die variabele in de 'selected' dataset. De boxplot komt tot stand doordat het algoritme achter de permutation importance meerdere iteraties doet, waardoor er dus meerdere daling in accuracy worden waargenomen. Deze worden in de boxplot samengevat. Hoe dichter de boxplot van een variabele bij nul komt, hoe minder het random forest model afhankelijk is van die variabele. De 10 variabelen met de hoogste permutation importance zijn geplott.


Puur vanuit de feature importances werd de conclusie getrokken dat variabelen die een telling bijhouden de meeste invloed werden toegekend. Echter vanuit de permutation importance blijkt dat er een andere volgorde ontstaat, want sommige niet-numerieke variabelen worden nu als belangrijker beschouwd.



Een voordeel van het gebruik van de permutation importance is dat het op meerdere soorten modellen toegepast kan worden in plaats van alleen op tree based modellen zoals random forests, wat geldt voor de feature importances. De drie figuren op deze pagina horen bij het random forest, SVM en KNN. De volgorde van de 10 variabelen met de hoogste permutation importance is voor geen van de drie modellen hetzelfde. Voor deze specifieke modellen kan er dus geconcludeerd worden dat er geen variabele is die duidelijk belangrijker is dan de andere. Er kan wel gekeken worden naar welke variabelen vaker naar voren komen. Dat zou een indicatie kunnen geven van de belangrijkheid van een variabele onafhankelijk van de modelkeuze.

Figuur 5: top 10 features met hoogste permutation importance per model. Nummering A, B, C van boven naar onder.

User\_creation\_tweet\_diff en tweet\_contains\_url zijn de enige twee variabelen drie keer in de top 10 voorkomen. User\_creation\_tweet\_diff staat daarbij zelfs alle drie de keren op de 6e plek of hoger. Tweet\_contains\_url vertoont bij het SVM weinig spreiding en staat daar zelfs bovenaan in de top 10 met een gemiddelde afname in accuracy van 2%. Echter bij de overige twee modellen verschilt de afname niet veel van nul en is de spreiding een stuk groter. Een groot deel van de variabelen in de top 10 van een van de modellen komt ook



een keer voor in een van de andere twee modellen. Voorbeelden hiervan zijn `user_is_verified`, `user_friends_count`, `tweet_sentiment`, `user_has_custom_timeline` en `twitter_android_user`.

Opvallend is dat de afnames in accuracy (x-as) bij het random forest veel kleinere waarden behaald dan bij de KNN en SVM. Dit betekent dat het model dus minder sterk afhankelijk is van een of een klein aantal variabelen, maar zijn voorspellingen baseert op een groter aantal features. Dit is wederom een indicatie dat de belangrijke variabelen sterk afhankelijk zijn van het type model.

Al met al kan er niet geconcludeerd worden dat er een aantal variabelen zijn die beduidend belangrijker zijn dan anderen. Het lijkt er namelijk op dat het type model een belangrijke rol speelt. Echter zijn er wel een aantal variabelen die vaker naar voren komen dan anderen zoals de `user_creation_tweet_diff` en `tweet_contains_url`. De totale afname in accuracy van deze variabelen blijft desalniettemin beperkt tot ongeveer 0.5 tot 2% als gevolg van de verschuiving die plaatsvindt bij de permutation importance.

## 10. Recall en Precision

### 10.1 Inleiding

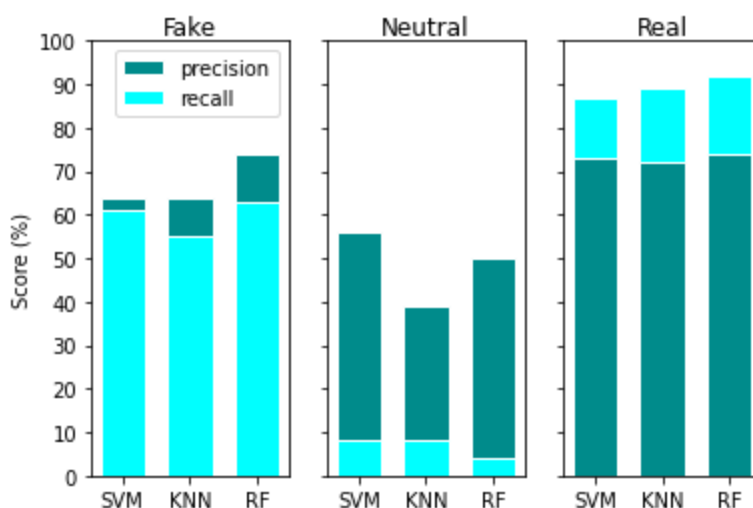
In het vorige hoofdstuk is een deel van het gedrag van machine learning algoritmen besproken aan de hand van de feature en permutation importance. Het gedrag van de algoritmen kan ook bestudeerd worden aan de hand van de precision en recall scores. Dit hoofdstuk bespreekt deze scores voor de machine learning algoritmes uit hoofdstuk 12 en een neurale netwerk. Het doel daarvan is om te bestuderen of het gebruik van metadata bij het classificeren van tweets systematisch dezelfde fouten maakt. Wanneer een dergelijk algoritme in de praktijk wordt ingezet is het namelijk van belang om te weten of het model bijvoorbeeld een neiging heeft om onterecht tweets als fake te labelen of dat hij juist nepnieuws mist.

### 10.2 Precision en Recall

Voor aan de analyse begonnen wordt, worden de betekenissen van precision en recall in onze context uitgelegd. De precision score betekent hoeveel procent van de voorspellingen daadwerkelijk hoort in de klasse die het algoritme voorspeld heeft. Een precision score van 100% betekent dat er geen False Positives zijn. Het model heeft dan geen voorspellingen onterecht als positief bestempeld. Dus stel een model heeft een 100% precision score van fake tweets, dan zijn alle tweets die hij voorspelt als fake ook daadwerkelijk fake. Echter kan het model daarnaast ook tweets voorspeld hebben als echt, terwijl ze daadwerkelijk fake zijn. Omdat de voorspelling niet fake is, komt deze fout niet tot uiting in de precision. Een 100% precision score betekent dus niet dat een model foutloos is. De hierboven beschreven fout komt wel tot uiting in de recall score. Op het gebied van fake tweets beantwoord deze score de volgende vraag: hoeveel procent van de daadwerkelijke fake tweets heb je goed voorspeld? Stel er zijn 100 fake tweets en een model maakt 80 keer een correcte voorspelling dat een tweet fake is, dan heeft hij er dus 20 gemist. Wanneer een model veel daadwerkelijke positieve klassen mist in zijn voorspellingen, dan is de recall laag.

### 10.3 SVM, KNN & RF Scores

In de onderstaande figuur staan de precision en recall scores per type tweet (fake, neutral, real) en per model (SVM, KNN, RF) op basis van alle metadata. De figuur dient niet gelezen te worden als een opstapeling, maar als twee staven die voor elkaar staan. Zo is bijvoorbeeld de recall score van het SVM model bij fake tweets 61% en ligt de precision score daar slechts een klein stukje boven met een score van 64%.

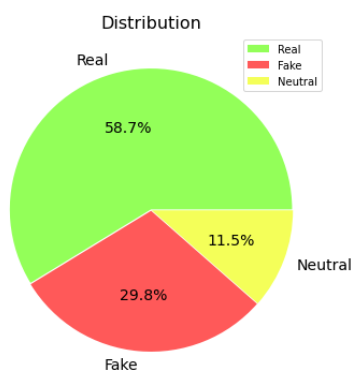


Figuur 6 : Precision en recall scores per modelsoort en type tweet.

Ten eerste valt op dat categorieën fake, neutral en real alle drie een ander beeld vertonen. Voor de fake tweets liggen de precision en recall scores relatief dicht bij elkaar, waarbij de precision score voor alle drie de modellen een aantal procentpunten hoger ligt. Op basis van dit beeld zou gesteld kunnen worden dat de modellen vrij goed gebalanceerd zijn op het gebied van recall en precision. Voor neutrale tweets is er van balans weinig sprake. De recall scores zijn namelijk extreem laag, namelijk onder de 10%. De precision scores die hieraan gekoppeld zijn, zijn een stuk hoger dan de recall scores, namelijk tussen de 40 en 60%, maar ten opzichte van de andere tweet categorieën blijven de precision scores ver achter.

De scores op de echte tweets zijn ten opzichte van de andere twee categorieën afwijkend. Het is namelijk de enige categorie waar de recall score hoger is dan de precision score. Ongeveer 70% van de tweets waarbij het label echt voorspeld is, is ook daadwerkelijk echt (precision score). De recall scores ongeveer 10 tot zelfs 20% procent hoger dan de precision scores. Dit betekent dat er veel van de daadwerkelijke echte tweets ook als echt bestempeld worden.

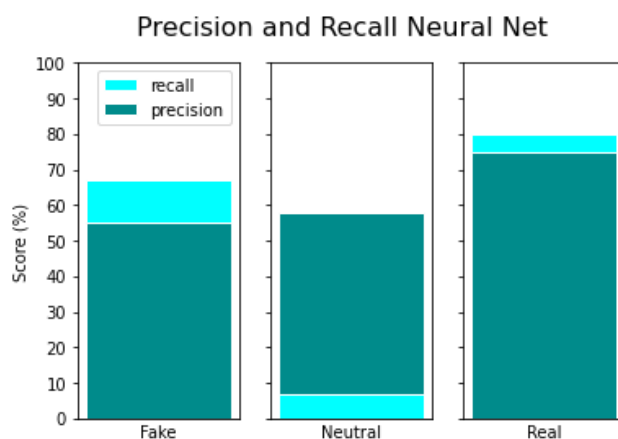
Een verklaring voor de uiteenlopende prestaties per klasse kan gevonden worden in de distributie van de tweets in de trainings dataset. In figuur 7 is deze verdeling te zien. Hieruit komt naar voren dat de dataset ruim 58% uit echte tweets bestaat, waardoor de modellen waarschijnlijk het beste presteren op deze categorie. De neutrale tweets zijn met ruim 11% veruit in de minderheid, waardoor het model weinig data heeft om op te trainen en dus slecht scoort. Het belang van een goede dataset wordt hiermee benadrukt.



Figuur 7: Verdeling an tweets in de trainingsdataset.

## 10.4 Neurale Network Scores

Een neurale netwerk vertoont een vergelijkbaar patroon als de eerder behandelde algoritmes bij de neutrale en echte tweets. Daarentegen is het gedrag bij de fake tweets anders. De vorige 3 modellen scoorden bij fake tweets een hogere precision dan recall score, maar bij het neurale netwerk is dat andersom. Een directe verklaring is hier niet voor gevonden. Echter illustreert het wel dat bij implementatie in de praktijk van classificatiemodellen op basis van metadata de modelsoort van belang kan zijn voor de type fouten die het model maakt en dat een maker zich hiervan bewust moet zijn.



Figuur 8: Precision en Recall per tweet soort van een neurale netwerk.

## 11. Resultaten op tekstuele inhoud

Het eerste team dat bezig was met het responsible AI project heeft twee datasets samengevoegd. De Constraint dataset bestaat alleen uit tweets en real fake grade. Vervolgens heeft het team een script gebruikt om de metadata op te halen van die tweets en deze samengevoegd met de data uit MisCov. Echter kon niet alle metadata opgehaald worden, omdat er tweets zijn verwijderd in de tussenperiode van making van de dataset en toen ze in gebruik zijn genomen door team 1.

Het totaal aantal rijen in de dataset die metadata en tekst bevat is 7838. Er is echter ook nog data beschikbaar waarvan alleen de tekst beschikbaar is en niet de metadata. Team twee heeft de tekstmodellen getraind op alle tweets. Hierbij hebben ze een dataset gebruikt van tweets met zo'n 14.000 rijen.

Het eerlijk vergelijken van onze resultaten met die van de vorige groepen gaat dus niet omdat de datasets verschillen. Zie [paragraaf 3.3](#) voor verdere informatie. Om deze reden zijn de modellen op basis van tweets opnieuw getraind met de dataset die in dit verslag gebruikt wordt.

### 11.1 Modellen

Qua tekst zijn er vier verschillende modellen getraind: Support Vector Machine, Passive-aggressive-classifier, (bi) Long Short-Term Memory, en een model genaamd BERT.

De SVM en PAC zijn machine learning classifiers. Op deze twee classifiers is een Grid Search gedaan om de optimale parameters te vinden. Alhoewel het trainen van deze modellen niet lang heeft geduurd, kan een Grid Search veel tijd in beslag nemen.

Long Short-Term Memory (LSTM) is een recurrent neurale netwerk dat in tegenstelling tot standaard feedforward neurale netwerken een feedback verbinding heeft. De LSTM is dus goed voor data met sequences. Voor tekstuele inhoud is daarom ook de LSTM gebruikt.

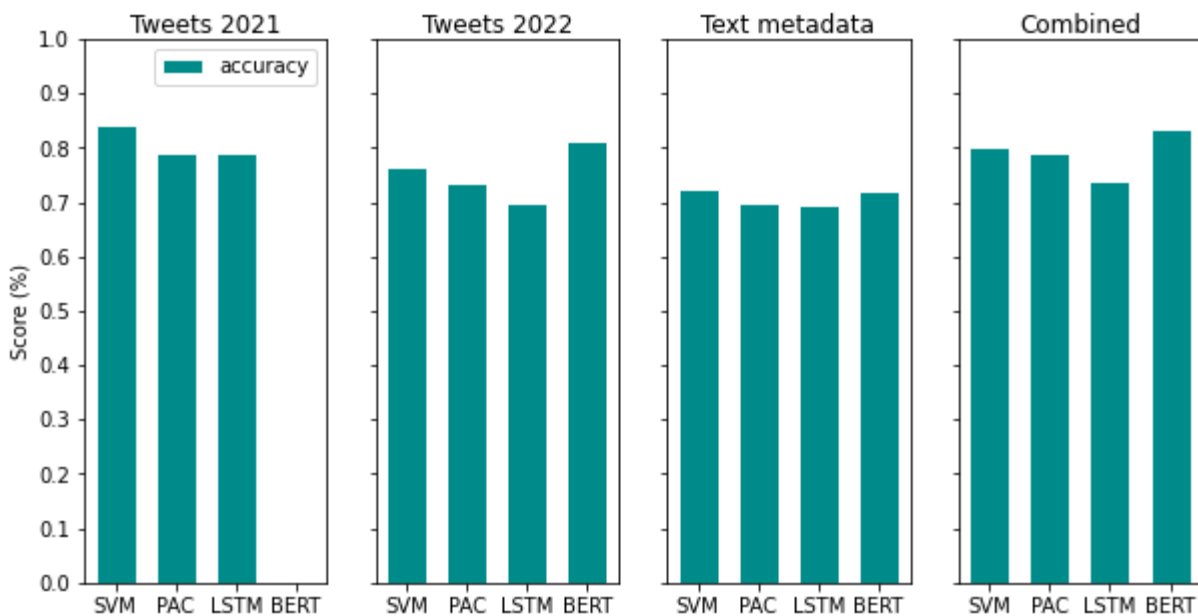
Het model bevat een embedding layer om elke woord in een vector te veranderen van een bepaald grootte. Daarnaast bestaat het model ook uit Bidirectional LSTM lagen. Er wordt gebruikt gemaakt van Bldirectional layers, omdat deze sequences van beide kanten leert. Na deze layer komt het gewone neurale netwerk dat bestaat uit Dense layers. De resultaten van de vorige team waren erg overfit. Daarom zijn er ook dropout layers toegevoegd om dit te voorkomen. De laatste layer is een softmax met drie output neuronen om te voorspellen of een tweet echt, nep of neutraal is.

Tot slot, het bert-model. Het BERT-model maakt gebruik van transformers. Transformers zijn 'attention' mechanismen die relaties in een tekst kunnen leren (Horev, 2018). De input hiervan is een reeks van tokens. De tokens bestaan uit o.a. input\_ids en attention\_masks.

Attention mask zijn reeks numerieke representaties van tokens (Horev, 2018). Deze bestaat uit nullen en enen. Input\_ids geven een unieke waarde aan elke token. Samen vormen ze de input voor de model. Net als de LSTM komt er na de BERT layer normale Dense layers met Dropout layers ertussen om overfitting te voorkomen.

## 11.2 Resultaten

Er zijn drie verschillende datasets met elkaar vergeleken. De modellen zijn getraind op tweet-data, tekst-metadata en een combinatie van de twee. Eerder in het verslag is er uitgelegd dat metadata vooral naar getallen is omgezet. De variabelen die tekst bevatten, zijn zo gelaten, zodat deze met de tekst-modellen getraind kunnen worden. Op dit moment bestaat tekst-metadata alleen uit username en description van een twitter-account. De combinatie bestaat uit de tweet-inhoud, username en description. In bijlage B is in [tabel 5](#) een kolom te zien met de naam 'Tweets 2021'. Deze is erbij gezet om het effect van data-grootte te zien.



Figuur 9: Barplots die de accuracy weergeven van verschillende modellen op de dataset

In figuur 9 zijn de resultaten van verschillende modellen te zien. De tweet-dataset komt bij alle classifiers hoger uit dan de metadata. Toch heeft de combinatie de grootste accuracy wat betreft tekstueel inhoud. Je zou verwachten dat tekst-metadata de accuracy van de tweets afremt, maar doordat er meer data beschikbaar is, wordt de accuracy ook hoger.

Ook is er te zien dat de tweets-2021 dataset het beter doet. Dat was al te verwachten, aangezien deze dataset twee keer zoveel informatie bevat als de dataset van 2022. BERT op tweets van 2021 is niet gerund, omdat deze te lang zou duren. Toch is het te verwachten dat de accuracy veel hoger uitkomt dan de dataset van 2022. Daarom mist er één balkje in figuur 9.

## 12. Combinatie tekst en metadata

Naast het testen van 'normale' modellen/netwerken was er het idee om meerdere Keras neurale netwerken te combineren in één netwerk om te kijken in hoeverre de accuraatheid verschilt ten opzichte van de op zichzelf staande netwerken.

Binnen Keras kan dit worden gedaan met de [Concatenate](#) laag. Deze laag verwacht als input één of meerdere outputs van andere netwerken en voegt ze dan samen tot één netwerk. Hierdoor wordt de uitkomst van het netwerk beïnvloed door meerdere netwerken met verschillende manieren van het leren van de data en kan het de uitkomst bevorderen of juist negatief beïnvloeden.

Deel van dit onderzoek was dan ook om te kijken of de los gebruikte neurale netwerken samen betere prestaties geven dan los van elkaar. De data die wij hebben gebruikt bestaat uit metadata en tekst. Hiervoor zijn twee verschillende neurale netwerken nodig geweest. Eentje om fake news te voorspellen aan de hand van metadata en een ander voor tekst.

Deze twee modellen werken totaal verschillend aan hun kant, voor metadata is het netwerk een simpel keras [Dense](#) netwerk met een aantal [Dropout](#) lagen. Voor het classificeren van tekst wordt gebruik gemaakt van een transformer genaamd [BERT](#). Deze twee netwerken werden samengevoegd tot één geheel d.m.v. de Concatenate laag, met daarna nog een klein Dense netwerk om de gecombineerde data van de twee te leren.

De gebruikte netwerken op zichzelf waren al voorgetraind en bruikbaar, dus voor het samenvoegen van de netwerken zijn er een paar aanpassingen gedaan. De lagen van de netwerken zijn op niet-trainbaar gezet omdat ze op dezelfde data al zijn getraind, dus het hertrainen ervan zou onnodig en tijdrovend zijn. Ook hadden beide netwerken een [Softmax](#) output laag voor de classificatie. Deze is voor de concatenate niet midden in het netwerk nodig, maar pas op het eind. Dus deze laag is er bij de individuele netwerken afgestript. Het enige in het gecombineerde netwerk wat dus werd getraind was de Concatenate laag samen met het Dense netwerk wat daarna volgde. De architectuur van het volledige netwerk is te zien in [Bijlage A Concatenated neurale netwerk](#).

### 12.1 Voorgaande keuzes

In de uiteindelijke situatie is het concatenate netwerk uitgekomen op het gebruik van een Dense netwerk (voor de numerieke metadata) i.c.m. meerdere BERT modellen (voor de tekst metadata). Voordat de transformer BERT in gebruik genomen is, werd het classificeren van de tekst metadata gedaan met LSTM netwerken. De resultaten hiervan waren niet heel spectaculair, vandaar de keuze om door te zoeken naar betere manieren van tekst classificeren.



## 12.2 Resultaten

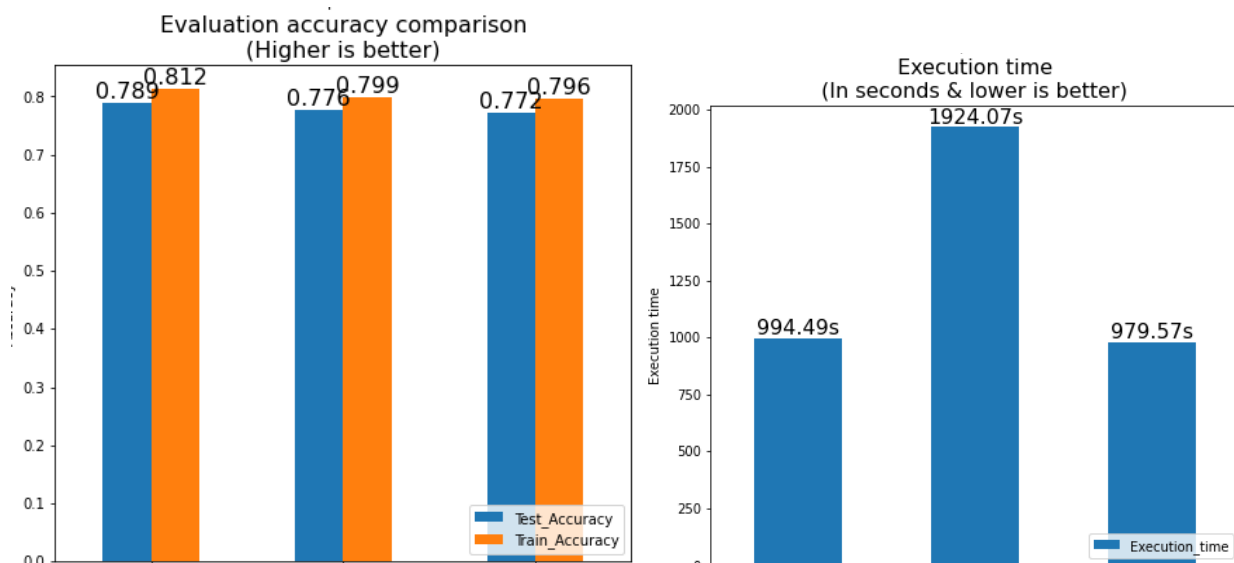
Het netwerk werd gedraaid met acht epochs met drie verschillende combinaties. De combinaties waren als volgt:

- Metadata + tekst-metadata (username + user description)
- Metadata + tweet
- Metadata + tekst-metadata (username + user description) + tweet

De eindresultaten zijn met elkaar vergeleken om te bestuderen hoe de verschillende combinaties ten opzichte van elkaar presteren. De verwachting was dat het grootste netwerk het best zou presteren, omdat deze de meest diverse data verwerkt en gebruikt tijdens het trainen. Een andere verwachting is dat het trainen van de netwerken lang duurt (lang relatief aan een standaard Dense netwerk) omdat het totale netwerk 335,621,737 parameters bevat.

De trainings- en testresultaten zijn in figuur 10 te vinden. De exacte percentages zijn nogmaals terug te vinden in tabel 5 van bijlage B. Van links naar rechts horen de scores bij de combinaties:

- (1) Metadata + tekst-metadata (username + user description)
- (2) Metadata + tekst-metadata (username + user description) + tweet
- (3) Metadata + tweet



Figuur 10: Accuracy en trainingstijd van de combinatie modellen

### 12.2.1 Samenvatting

De opgesomde trainingstijd met GPU was een uur. Van dat uur nam het grootste netwerk (metadata + userinfo + tweet) het grootste gedeelte in beslag, namelijk een half uur. Dit kwam overeen met de verwachting.

Ook blijkt dat de accuraatheid scores nagenoeg gelijk zijn. De train accuraatheid ligt tussen de 79.6% en 81.2%. Het verschil is dus slechts 1.6%. Het verschil in de test accuraatheid is net iets meer, 1.7%. Als we de accuraatheid scores en de gebruikte tijd samen nemen is er goed te zien welk netwerk de slechtste resultaten gaf, en dat is het grootste netwerk. Dit netwerk van 3 datasets had 32 minuten nodig om tot dezelfde resultaten te komen als een netwerk met maar 2 datasets. Maar het is wel nodig om te zeggen dat de netwerken op maar 8 epochs zijn getraind. De reden voor 8 epochs is omdat het trainen van dit netwerk gezien de resultaten best lang duurt.

Voorlopig blijkt dat het grote netwerk overbodig is, omdat er waarschijnlijk onvoldoende genoeg verbanden kunnen worden getrokken tussen de metadata, tweets en de tekst-metadata om beter te scoren dan de overige combinaties. De andere kleinere netwerken scoren namelijk nagenoeg hetzelfde op accuraatheid maar de executie tijd is beduidend minder.

### 12.2.2 Vergelijking met de alleenstaande netwerken

in hoofdstuk 13.2 waren de resultaten van de gecombineerde netwerken te zien. De losse netwerken op zichzelf hadden hele andere scores:

- metadata netwerk: **69%**
- BERT userinfo: **72%**
- BERT tweet: **81%**

Het moment dat de netwerken werden gecombineerd kwamen er andere resultaten uit:

- metadata + userinfo: **78.9%**
- metadata + tweet: **77.2%**
- metadata + userinfo + tweet: **77.6%**

Wat de resultaten hierboven laten zien is dat het tweet BERT model de hoogste accuraatheid op zichzelf had van 81%, maar zodra deze wordt gecombineerd met het metadata netwerk dan gaat de totale accuraatheid achteruit met ongeveer 3%. Daarnaast blijkt ook dat de metadata van de tweets (userinfo en andere omliggende data) de scores iets naar onder trekken.

## 13. Ensemble Learning

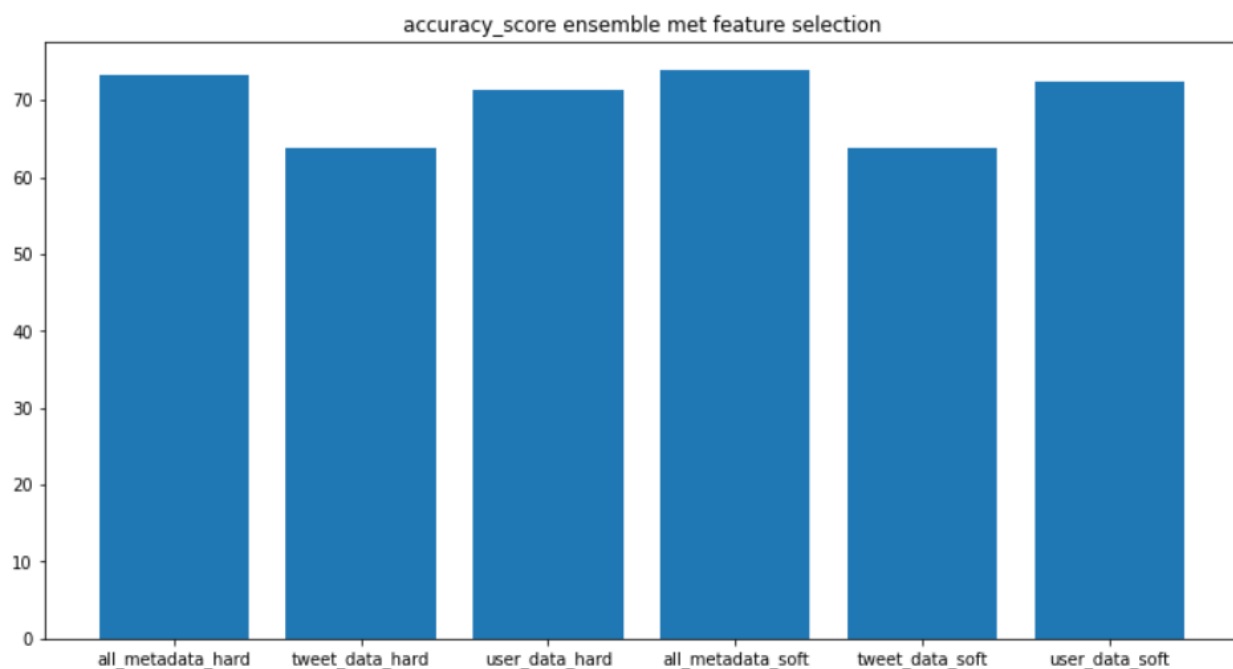
De KNN, RF en SVM modellen zijn samengevoegd door middel van ensemble learning. Het doel hiervan is om de voorspellingen van de verschillende modellen te combineren. Ensemble learning zou goed kunnen werken wanneer de verschillende modellen verschillende systematische fouten maken. Het combineren van de modellen zou ervoor kunnen zorgen dat de verschillende modellen elkaars fouten compenseren en dat de algemene uitkomst daardoor verbeterd wordt. Om het goed te testen zijn zowel soft- als hard voting gebruikt. Bij hard voting gaat het om het aantal stemmen. Als de KNN classificeert dat een tweet nep is, maar de andere twee classifiers detecteren dat het echt nieuws is, dan wordt het geclassificeerd als echt nieuws. Echt nieuws heeft immers de meeste stemmen.

Bij soft voting wordt de argmax terug gegeven van de voorspelde predicties. Argmax geeft de index van de hoogste getal terug. Elke classifier krijgt ook een gewicht. Voor dit model zijn alle gewichten op 1 gezet. Voorbeeld van soft voting (1.11. *Ensemble Methods*, 0) waarin  $w_1$ ,  $w_2$  en  $w_3$  de gewichten zijn.

classifier	class 1	class 2	class 3
classifier 1	$w_1 * 0.2$	$w_1 * 0.5$	$w_1 * 0.3$
classifier 2	$w_2 * 0.6$	$w_2 * 0.3$	$w_2 * 0.1$
classifier 3	$w_3 * 0.3$	$w_3 * 0.4$	$w_3 * 0.3$
weighted average	0.37	0.4	0.23

in dit voorbeeld zou dus class 2 de gekozen uitkomst worden.

De resultaten van het model zijn hieronder te zien.



*Figuur 11: Accuracy scores m.b.v. ensemble learning*

Te zien is dat de waarde op de verschillende datasets anders presenteren in vergelijking met de lossen classifiers. Wanneer gebruik gemaakt wordt van soft voting op alle metadata blijkt dat er een betere score behaald wordt ten opzichte van de individuele modellen. De user data en tweet data scores worden niet verbeterd door gebruik te maken van hard of soft voting.

## 14. Attention

### 14.1 Wat is Attention?

Attention, in het Nederlands aandacht, is in machine learning een relatief nieuwe term. In een psychologische context is de betekenis van attention: het cognitieve proces van selectief concentreren op één aspect van de omgeving terwijl andere dingen worden genegeerd (Psychology Wiki. z.d.).

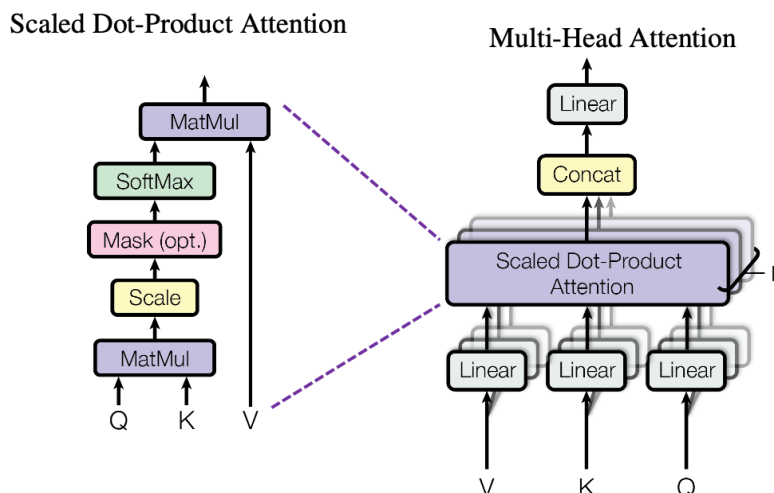
Het principe van attention is tegenwoordig in machine learning ook terug te vinden. Het wordt vooral gebruikt bij spraakherkenning en het automatisch vertalen van teksten. Waar voorheen LSTM de beste modellen waren voor zulke taken, volgden recentelijk LSTM modellen met attention layers en inmiddels zijn de LSTM modellen hiervan ook gestript (P. Wiggers, persoonlijke communicatie, 3 juni 2022).

Attention layers worden dus gebruikt om te bepalen op welke delen van bijvoorbeeld een tekst het model zijn aandacht moet leggen. Door meerdere van dit soort lagen te combineren op een vergelijkbare manier als de filters in een CNN netwerk ontstaan multi-head Attention lagen, waarbij elke head zijn eigen focus heeft (net zoals elke filter in een CNN). Door meerdere van deze multi-head Attentions met elkaar te combineren ontstaan een model genaamd een Transformer. Het concept van attention is nog vrij nieuw en is daarom lastig om volledig te begrijpen. Echter is het nog steeds interessant om te testen.

### 14.2 Toepassen van Attention

Attention layers hebben drie parameters: de query (Q), de key (K) en de value (V). De query is hetgeen waar het model in geïnteresseerd is, bijvoorbeeld een tekst. Deze gaat samen met de key het model in, waar het meerdere stappen doorloopt om uiteindelijk te eindigen in een SoftMax laag. Deze laag kijkt voor elk element in de query waar hij mee samenhangt. De key en de value zijn de delen waar de aandacht op gericht wordt.

Voor dit onderzoek is het interessant om te bestuderen hoe de uitkomsten veranderen bij verschillende vormen van de key. De query zal de tekst van de tweet zijn, maar de key kan veranderd worden. Er worden drie vormen getest, namelijk alle metadata, de user metadata en de tweet metadata. De figuur hieronder geeft een overzicht van hoe een Attention layer er uit kan zien. Een Multi-Head Attention bestaat vervolgens uit een opstapeling van Attention layers.



Figuur 12: Schematische weergave van een Attention layer en een Multi-Head Attention (Vaswani et. al, 2017)

### 14.3 Self Attention CyberZHG & Bi-LSTM

Het eerst model dat getest is, is met de self attention layer implementatie voor keras (CyberZHG, n.d.), deze wordt getest onder de text modellen in *attention-bilstm.ipynb*. Echter wordt geconstateerd dat dit niet een optimale implementatie is voor het self attention mechanisme. Wanneer deze layer wordt gebruikt zal het model niet meer informatie uit de data halen dan bij de attention implementatie in het concatenate netwerk waar de verschillende soorten data worden gecombineerd. De accuracy blijft dan ook hangen bij 68% voor de 2021 dataset.

### 14.4 Attention layer

Een betere implementatie van de attention layer is die in keras zelf. Deze wordt getest onder de modellen in *attention.ipynb*. Hier wordt gebruik gemaakt van een convolutional netwerk, voor de query en value in de attention layer. Verder bestaat het model uit een Dropout layer en een aantal Dense layers (zie [Bijlage C Attention netwerk](#)).

Voor het trainen van dit netwerk hebben wij uiteraard dezelfde dataset gebruikt die werd gebruikt bij het concatenate netwerk uit *concatenate.ipynb*. Dus de dataset:

- met de pure metadata;
- text metadata (username, user description);
- de tweet zelf

Omdat bij de attention layer drie datasets als input meegegeven kunnen worden (minimaal twee), waarbij iedere input een ander pad aflegt door het netwerk en dus andere resultaten kan produceren, zijn er zoveel mogelijk combinaties getest. Dus bijvoorbeeld bij de eerste sessie bij dit netwerk is de query input de metadata en de value input de tweet

zelf. Bij de tweede sessie is het omgedraaid, de query input is dan tweet en de value input de metadata.

Uiteindelijk zijn er zeven verschillende combinaties uitgekomen:

- 1) userinfo & tweet
- 2) metadata & userinfo & tweet
- 3) metadata & userinfo
- 4) userinfo
- 5) metadata & tweet
- 6) tweet
- 7) metadata

Bij de bovenstaande combinaties geldt dat bij gebruik van een enkele dataset dat de query en de value parameter door diezelfde set wordt ingevuld. Bij de combinaties met twee datasets (bijv. metadata-tweet) wordt de query parameter ingevuld door metadata, en de value parameter door de andere dataset.

Bij de combinaties met drie datasets komt de key parameter erbij. Dan is de volgorde: query, key, value. Er zijn dan dus drie inputs. Iedere combinatie is met 30 epochs gedraaid en met een batch size van 32.

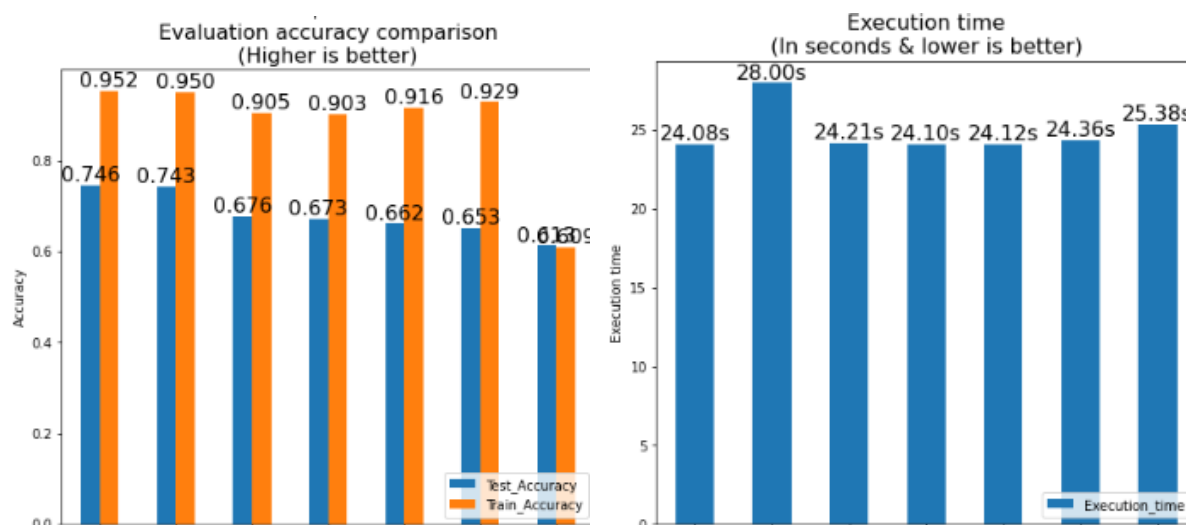
#### 14.4.1 Resultaten

De resultaten staan in de onderstaande figuur met de combinaties op de onderstaande volgorde van links naar rechts:

- 1) userinfo-tweet
- 2) metadata-userinfo-tweet
- 3) metadata-userinfo
- 4) userinfo
- 5) metadata-tweet
- 6) tweet
- 7) metadata

Het trainen van alle combinaties duurde in totaal 174 seconden, dus bijna 3 minuten. De beste score die eruit kwam was een test accuraatheid van 74.6% met een executietijd van 24 seconden. Dit was het attention netwerk met maar twee inputs: userinfo-tweet met userinfo als query en tweet als value.

De resultaten waren nagenoeg gelijk aan het netwerk met drie inputs: metadata-userinfo-tweet (in de volgorde query-key-value), die een test accuraatheid van 74.3% behaalde. De andere netwerken kwamen de grens van 70% niet over. Alle percentages zijn wederom terug te vinden in Bijlage B (tabel nummer 6).



Figuur 13: Training en test accuracy scores (a) en trainingstijden (b) van de attention netwerken.

#### 14.4.2 Vergelijking met Concatenate netwerk

Het is moeilijk om het attention netwerk en het concatenate netwerk met elkaar te vergelijken omdat deze op hele verschillende manieren werken.

Het concatenate netwerk maakt gebruik van voorgetrainde netwerken. Waarbij voor het metadata netwerk (wat erin wordt gebruikt) alle gewichten zijn voorgetraind op die data. Hetzelfde geldt voor de BERT modellen die worden gebruikt. Het enige moment waar de data samenkomt is op het eind van het totale netwerk bij een Concatenate laag die de samengevoegde data doorspeelt naar een relatief klein Dense netwerk waar het verbanden tussen de verschillende data probeert te leggen.

Bij het attention netwerk probeert het netwerk verbanden te leggen tussen de output van de attention laag (query en mogelijk key input) en de value input. De resultaten van de attention laag en de value input worden samengevoegd door een concatenate laag en de resultaten worden geleerd door een Dense netwerk dat volgt.

Uiteindelijk is het moeilijk om de resultaten op gelijke grond te vergelijken, het attention netwerk is op 30 epochs met een batch size van 64 getraind. Het concatenate netwerk met 8 epochs en een batch size van 128. Het concatenate netwerk deed het qua resultaten wel beter, de hoogste accuraatheid was 78.9%, voor het attention netwerk is dit 74.6%. Attention deed het met de executietijd wel stukken beter, namelijk slechts 24 seconden ten opzichte van de 994 seconden die het Concatenate netwerk erover deed. Het Concatenate netwerk duurde dus 41x zo lang terwijl het een stuk minder epochs heeft gerund. Daarentegen is het concatenate netwerk wel een heel stuk complexer, het bestaat overigens uit 2 á 3 (1x neuraal net, 2x BERT) compleet functionele netwerken.

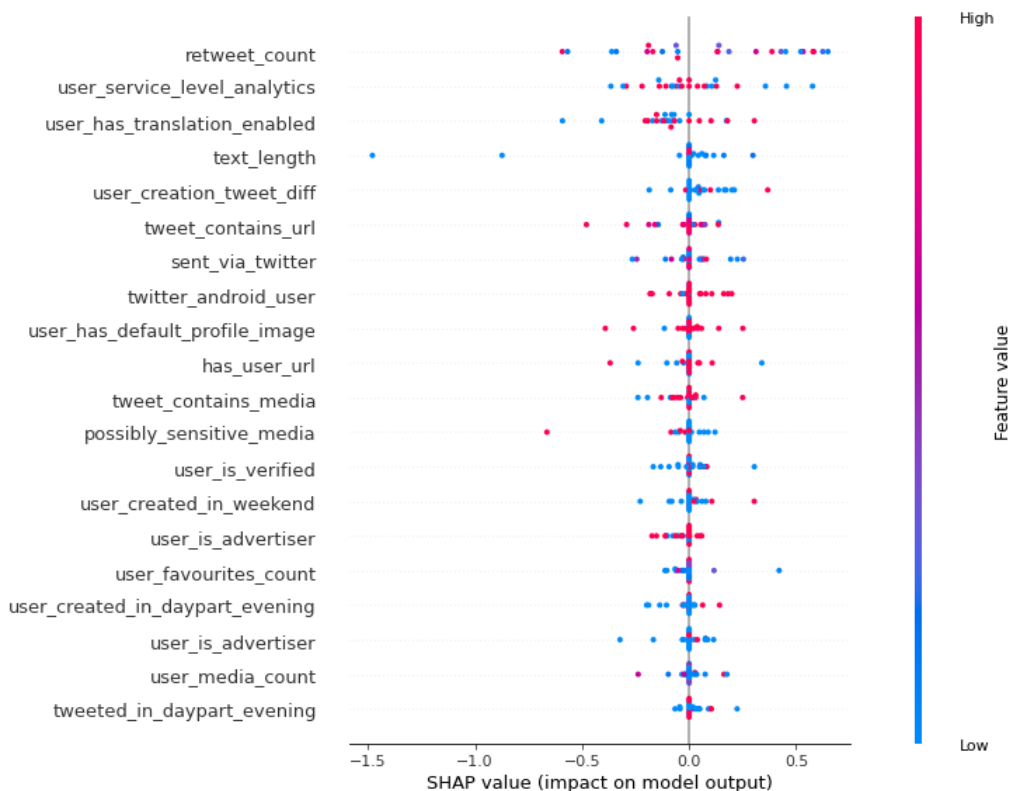


## 15. Explainable AI

SHAP is een explainable AI (XAI) methode die werkt door uit te leggen wat de bijdrage is van elke feature in het model. Shap gebruikt een eigen variant van de Shapley values. Deze worden gebruikt om te berekenen hoeveel bijdrage een waarde heeft in het totaal. Wat shap anders maakt dan Shapley is dat door KernelSHAP er een uitleg gegeven kan worden voor individuele voorspellingen. In SHAP wordt elke feature als een losse speler gezien van een team (de hele dataset) en elke speler heeft een eigen bijdrage aan het team en is ook afhankelijk van het hele team.

De vorige teams hebben uitgebreid aandacht besteed aan Explainable AI methodes. De opdrachtgever heeft de voorkeur uitgesproken om in dit verslag aandacht te besteden aan iets nieuws: de attention netwerken. Er is dus weinig aandacht besteed aan Explainable AI in dit verslag. Echter om het niet volledig buiten beschouwing te laten is er wel wat aandacht aan geschonken.

Het hieronder te vinden SHAP plot is gerund op een SVM metadata model.



Te zien zijn de features die dit model het belangrijkste vond, waarvan boven naar beneden de features op belangrijkheid staan en de punten zijn hoe belangrijk de individuele samples ze vonden en hoe dat bijdraagt aan de real; of fake grade.

Het onderliggende plaatje laat de SHAP output zien van een sample en hoe belangrijk elk feature was voor de output.

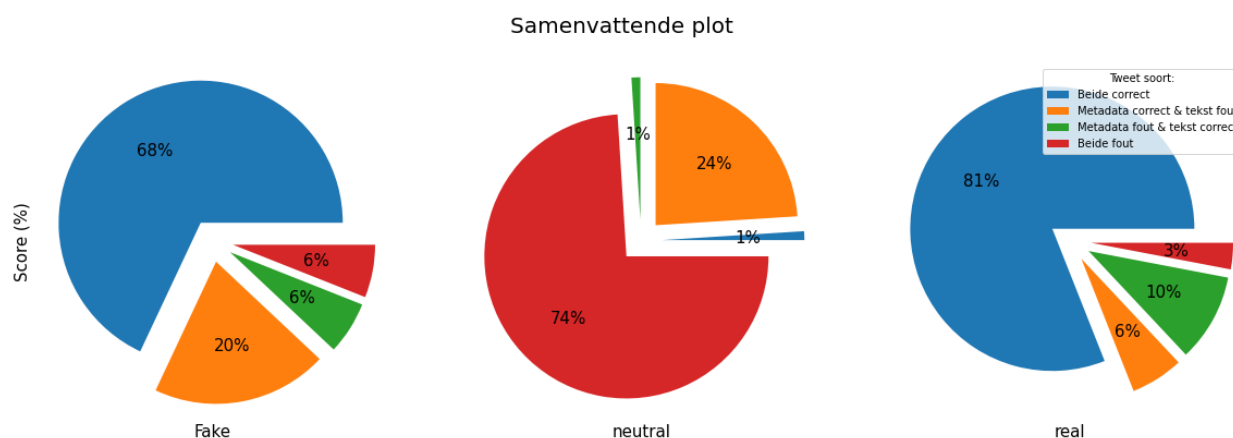


In het onderliggende plaatje zijn de belangrijke features te zien voor het text model.

## 16. Samenvattende plot

Dit hoofdstuk bespreekt een vergelijking van de resultaten van twee verschillende modellen. Het op tekst gebaseerde BERT model wordt vergeleken met het neurale netwerk (NN) op metadata. Het doel van deze vergelijking is om te achterhalen hoe de classificatie resultaten op basis van tekst en metadata per tweet categorie zich tot elkaar verhouden. De onderstaande figuur geeft per tweet type (fake, neutral, real) vier categorieën aan van de fractie tweets die:

1. door beide modellen correct voorspeld is (dus twee afzonderlijke correcte voorspellingen)
2. door het NN correct voorspeld is, maar door het BERT model incorrect voorspeld is
3. door het NN incorrect voorspeld is, maar door het BERT model correct voorspeld is
4. door beide modellen incorrect voorspeld is (dus twee afzonderlijke incorrecte voorspellingen)



Figuur 15: Samenvattende plot over de classificatie resultaten gebaseerd op metadata en tekst afzonderlijk van elkaar.

Allereerst valt op dat de fracties met twee correcte voorspellingen bij de neutrale tweets extreem klein is ten opzichte van de overige twee tweet typen. Een soortgelijk patroon is eerder ook bij de recall en precision geconstateerd. Dit zou te maken kunnen hebben met de verdeling van de tweets in de dataset (zie figuur 7). Daarnaast geldt dat de echte tweets het enige type is waarbij het groene vlak groter is dan het oranje vlak. Dit houdt in dat de fractie tweets waarbij metadata fout en tekst correct voorspeld groter is dan waarbij dit andersom is. Bij neppe en neutrale tweets zijn de oranje vlakken namelijk aanzienlijk groter dan de groene vlakken. Al met al lijkt het erop dat op basis van metadata een grote fractie correcte voorspellingen gemaakt kan worden wanneer er genoeg data beschikbaar is.

## 17. Conclusie

Tijdens dit project is er onderzoek gedaan naar de invloed die metadata heeft op het detecteren van nepnieuws op basis van Covid-19 tweets. Met name de invloed van metadata ten opzichte van en in combinatie met de tweet inhoud is naar voren gekomen. Dit hoofdstuk bespreekt alle bevindingen en conclusies. De resultaten van dit onderzoek zijn te reproduceren middels de code te runnen uit onze [gitlab repository](#).

### 17.1 Permutation Importance

Er is onderzocht welke metadata features het meeste invloed uitoefenen op de uitkomsten van de machine learning modellen. In eerste instantie is de feature importance hiervoor gebruikt. Echter blijkt deze methode een sterke bias te hebben richting de variabelen die een telling bijhouden zoals het aantal hashtags of retweets. De permutation importance geeft een betrouwbaardere weergave. Er is geen eenduidige conclusie te trekken over welke features het meeste van belang zijn. Echter zijn er wel twee variabelen die in alle drie de machine learning modellen in de top 10 features met hoogste permutation importance staan. Dit zijn de `user_creation_tweet_diff` (het tijdsverschil tussen het aanmaken van de account en het plaatsen van de tweet) en `tweet_contains_url` (of een tweet een url bevat). De `user_creation_tweet_diff` is een door onze groep in elkaar gezet. Mocht er diepgaander onderzoek door bijvoorbeeld twitter gedaan willen worden dan zou het handig kunnen zijn om als bedrijf deze feature zelf in de metadata op te nemen.

### 17.2 Recall & Precision

Bij het gedetailleerder bestuderen van het gedrag van de machine learning algoritmes zijn de recall en precision scores gebruikt per type tweet: nep, neutraal en echt. Er blijkt dat de precision en recall voor fake tweets relatief dicht bij elkaar liggen: beide tussen de 60% en 70%, waarbij de precision score slechts een aantal procentpunten hoger is dan de recall. Neutrale tweets laten een heel ander patroon zien: extreem lage recall scores, namelijk onder de 10% en precision scores van tussen de 40 en 60%. Echte tweets laten weer een ander patroon zien. Het is namelijk de enige categorie waarin de recall hoger is dan de precision. Ongeveer 70% van de tweets waarbij het label echt voorspeld is, is ook daadwerkelijk echt (precision score). De recall scores ongeveer 10 tot zelfs 20% procent hoger dan de precision scores. Dit betekent dat er veel van de daadwerkelijke echte tweets ook als echt bestempeld worden. Een verklaring voor de uiteenlopende prestaties per klasse kan gevonden worden in de distributie van de tweets in de trainings-dataset. In figuur 7 is deze verdeling te zien. Hieruit komt naar voren dat de dataset ruim 58% uit echte tweets bestaat, waardoor de modellen waarschijnlijk het beste presteren op deze categorie.

## 17.3 Tekst-data

Qua tekst zijn er vier verschillende modellen getraind: Support Vector Machine, Passive-aggressive-classifier, (bi) Long Short-Term Memory, en een model genaamd BERT. Deze zijn op verschillende datasets getest: tweet-data, tekst-metadata en een combinatie van de twee. Een belangrijke conclusie is dat bij alle classifiers de tweet-dataset hoger scoort dan de tekst-metadata en dat de combinatie van de tweet inhoud en de tekst metadata het beste scoort. Aangezien een combinatie van de twee meer data bevat, geeft het ook betere resultaten. De BERT-model met de combinatie van tweets en tekst-metadata als dataset scoort het hoogst.

## 17.4 Concatenate

Om te onderzoeken wat het effect is van het combineren van tekst-data en metadata is gebruik gemaakt van een neurale netwerk dat verschillende afzonderlijke netwerken samenvoegt. Dit is gedaan door gebruik te maken van een Concatenate layer uit keras. Er zijn drie combinaties getest:

- Metadata + tekst-metadata (username + user description)
- Metadata + tweet
- Metadata + tekst-metadata (username + user description) + tweet-inhoud

Er blijkt dat de test en trainings accuracy scores van nagenoeg gelijk zijn aan elkaar. De test accuracy scores liggen allemaal tussen de 77 en 79%. Verder blijkt dat het netwerk met alle drie de subdatasets als input ruim twee keer zo veel tijd in beslag neemt tijdens het trainen, waardoor dit netwerk overbodig lijkt. Het Bert model dat enkel tweet inhoud als input heeft scoort 81% als accuracy. Uit het Concatenate netwerk blijkt dat de accuracy met ongeveer 3% daalt wanneer metadata aan de input data wordt toegevoegd. Ook daalt de accuracy wanneer er tekst-metadata en de tweet inhoud wordt toegevoegd.

Een belangrijke conclusie is dus dat op basis van het Concatenate netwerk tweet-inhoud op zichzelf de beste voorspellingen maakt. Door een vorm van metadata (tekst of 'normaal') extra mee te geven als input wordt de accuracy score slechter.

## 17.5 Ensemble learning

Een andere methode die gebruikt is om de combinatie van modellen te onderzoeken is ensemble learning. Hierbij was de hoop dat de systematische fouten van de afzonderlijke modellen door ze te combineren elkaar zouden compenseren. Echter blijkt dit niet het geval. De RF, SVM en KNN modellen zijn op twee manieren gecombineerd: hard-voting en soft-voting. Wanneer gebruik gemaakt wordt van soft voting op alle metadata blijkt dat er een betere score behaald wordt ten opzichte van de individuele modellen. De user data en tweet data scores worden niet verbeterd door gebruik te maken van hard of soft voting.

## 17.6 Attention

Een alternatieve methode om tekst en metadata met elkaar te combineren is het gebruik van attention layers. Allereerst is de self attention layer voor keras van CyberZHG gebruikt. Op onze dataset raakte deze implementatie snel overfit en scoort slechts 68% accuracy. Vervolgens zijn verschillende query-key-value combinaties geprobeerd als input voor een model met een attention layer (uit keras). Hieruit blijkt dat de volgende combinaties het beste scoren met een score van rond de 75% accuracy: userinfo & tweet-inhoud en de combinatie metadata & userinfo & tweet-inhoud. De overige combinaties komen niet boven de 68% uit. De trainingstijden van de verschillende combinaties zijn nagenoeg gelijk. Echter ten opzichte van het Concatenate netwerk is er een groot verschil. Het attention netwerk traint 41x sneller terwijl het met 30 epochs is gerund en het Concatenate netwerk slechts met 8. Door de complexiteit van het Concatenate netwerk scoort dit netwerk wel beter dan het beste attention netwerk.

## 17.7 Algemene conclusie

Metadata en tekst data zijn op verschillende manieren aan bod gekomen in dit verslag. De hoofdvraag van dit onderzoek was: In hoeverre is nepnieuws op Twitter te detecteren aan de hand van Twitter metadata in vergelijking met de detectie op grond van de tekstuele inhoud van een tweet?

Op basis van de resultaten van dit verslag is er geconcludeerd dat tijdens het detecteren van nepnieuws met metadata een grens bereikt wordt van 75%. Alle verschillende manieren die getest zijn hebben namelijk geen hogere score opgeleverd. De training accuracy kan wel hoog op lopen, maar het lijkt alsof er een barrière is bij 75% test accuracy.

Een andere overkoepelende conclusie is dat tekst data beter presteert dan metadata of een combinatie van de twee. Echter is het verschil niet al te groot. Het beste tekst-model is een BERT model met een score van 81%, terwijl het beste metadata model een soft voting ensemble model was met een score van 74%. De beste combinatie komt voort uit het Concatenate model met een score van 79%.

Doordat metadata- en tekstclassificatie resultaten relatief sterk met elkaar gecorreleerd zijn is het advies meer aandacht te geven aan metadata vanwege minder tijdsafhankelijke kenmerken.

## 18. Aanbevelingen

In dit hoofdstuk worden enkele aanbevelingen genoemd voor vervolgonderzoeken en het gebruik van metadata. De eerste aanbeveling is dat de `user_creation_tweet_diff` verder onderzocht kan worden en standaard in bijvoorbeeld de metadata van Twitter wordt opgenomen wanneer daadwerkelijk blijkt dat deze variabele helpt bij het detecteren van bijvoorbeeld het gebruik van bots.

Daarnaast is het vinden van een nieuwe dataset van belang om een extra test op te kunnen uitoefenen. Het probleem is echter dat betrouwbare datasets omtrent Twitter nepnieuws lastig te vinden zijn. Een bruikbaar onderzoek zou dus kunnen zijn om een nieuwe dataset samen te stellen. Het zou daarbij ook interessant zijn om de scope te vergroten naar andere sociale media zoals Facebook of Reddit. Naast andere sociale media platformen zou ook gekeken kunnen worden naar andere thema's in plaats van alleen Covid-19.

In dit verslag zijn verschillende attention netwerken getest. Hieruit is naar voren gekomen dat ze relatief snel trainen en vrij goed scoren. Echter is wegens tijdsgebrek vooral tijd besteed aan het werkend krijgen van het netwerk in plaats van het optimaliseren ervan. Een vervolgonderzoek zou aandacht kunnen besteden aan het optimaliseren van de parameters en het uitproberen van verschillende architecturen.

In dit onderzoek zijn verschillende modellen samengevoegd in een Concatenate netwerk. Het attention netwerk is hier echter niet in geprobeerd, omdat het attention netwerk pas aan het eind van het onderzoek in elkaar is gezet. Het zou interessant kunnen zijn om het attention netwerk in een Concatenate netwerk te stoppen en te combineren met andere modellen.

Een andere aanbeveling zou kunnen zijn om een model te ontwikkelen dat metadata en tekst als input heeft en op basis van twee gewichten de verhouding tussen houdbaarheid en prestatie aan te kunnen passen. De gewichten bepalen dan de mate waarin de voorspelling van metadata of tekst meegewogen wordt in de eind voorspelling. Het zou dus een soort ensemble learning model kunnen zijn. De gebruiker zou dan het gewicht van de metadata op 0 en van tekst op 1 zetten wanneer houdbaarheid compleet onbelangrijk is en een optimale prestatie de wens is. Wanneer het belang van houdbaarheid in het model toeneemt zou de gebruiker of ontwikkelaar ervoor kunnen kiezen om het gewicht van de metadata in het model te verhogen.

## 19. Bibliografie

CyberZHG. (n.d.). *keras-self-attention* · PyPI. PyPI. Retrieved June 8, 2022, from

<https://pypi.org/project/keras-self-attention/>

Flietstra, J., Geerts, J., Hondema, T., & van Ommeren, M. (2021, 6). *Toelichting bij automatische herkenning van nepnieuws: een toepassing voor COVID-19*. [HVA]. Amstelcampus.

Grelf, B. (2018, 11 17). How easily you can be identified by publicly available Twitter metadata.

<https://cliqz.com/en/magazine/how-easily-you-can-be-identified-by-publicly-available-twitter-metadata>

Horev, R. (2018, 11 17). BERT Explained: State of the art language model for NLP.

<https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270>

ICT Portal. (2018, November 20). *Metadata classificeren en contextualiseren informatie*. ICT Portal. Retrieved June 8, 2022, from <https://www.ictportal.nl/ict-lexicon/metadata>

Keras. (2022). *Concatenate layer*. Keras. Retrieved June 8, 2022, from

[https://keras.io/api/layers/merging\\_layers/concatenate/](https://keras.io/api/layers/merging_layers/concatenate/)

Molnar, C. (2022). *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*. Retrieved June 8, 2022, from

<https://christophm.github.io/interpretable-ml-book/feature-importance.html>

Perez, B. (2018, 3). You are your Metadata: Identification and Obfuscation of Social Media Users using Metadata Information. *AAAI International Conference on Web and Social Media*. researchgate.



<https://www.researchgate.net/publication/324055375> You are your Metadata Identification and Obfuscation of Social Media Users using Metadata Information

Psychology Wiki. (z.d.) *Introduction to attention*. Geraadpleegd op 2 juni 2022, van

[https://psychology.fandom.com/wiki/Introduction\\_to\\_attention](https://psychology.fandom.com/wiki/Introduction_to_attention)

Redactie. (2017, June 14). Met nepnieuws verkiezingen manipuleren? Kinderspel, blijkt uit studie. *AD.nl*.

<https://www.ad.nl/buitenland/met-nepnieuws-verkiezingen-manipuleren-kinderspel-blijkt-uit-studie~ab7aa349/>

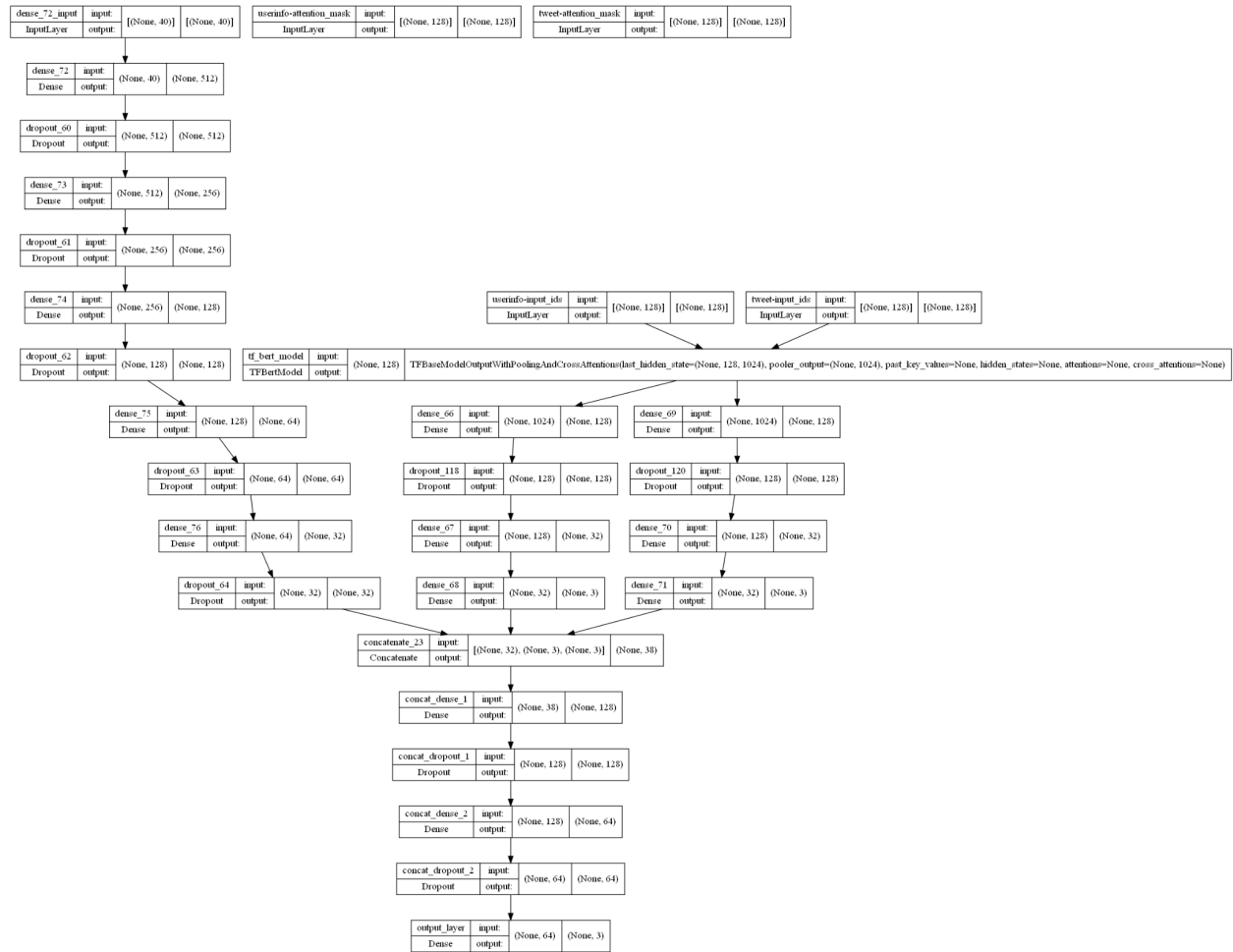
Redactie. (2020, August 21). Twitter pakt Nederlandse 'corona-trollen' aan. *RTL Nieuws*.

<https://www.rtlnieuws.nl/tech/artikel/5178584/twitter-trollen-nepnieuws-coronaviruss-nederland>

Vaswani, A., Shazeer, N., Parmar, N., et. al. (2017) *Attention Is All You Need*. geraadpleegd van <https://arxiv.org/abs/1706.03762> op 2 juni 2022

## 20 Bijlagen

### 20.1 Bijlage A Concatenated neurale netwerk



## 20.2 Bijlage B Tabellen

**Tabel 1: Accuracy scores op dataset zonder feature selectie (H7.2)**

model	alle data	tweet data	user data
<i>Support Vect. Mach.</i>	73%	66%	68%
<i>K-Nearest Neighb.</i>	69%	64%	70%
<i>Random Forest</i>	73%	69%	72%
<i>Ensemble hard voting</i>	74%	65%	71%
<i>Ensemble soft voting</i>	73%	66%	71%

**Tabel 2: Accuracy, recall en precision scores op dataset met feature selectie (H8)**

model	alle data	tweet data	user data
<i>Support Vect. Mach.</i>	70%	69%	72%
<i>K-Nearest Neighb.</i>	69%	66%	71%
<i>Random Forest</i>	73%	72%	75%
<i>Ensemble hard voting</i>	73%	64%	71%
<i>Ensemble soft voting</i>	74%	64%	72%

**Tabel 3: Precision en recall scores op alle metadata per type tweet (H11)**

Model	<i>fake (-1)</i>		<i>Neutral (0)</i>		<i>Real (1)</i>	
	<i>precision</i>	<i>recall</i>	<i>precision</i>	<i>recall</i>	<i>precision</i>	<i>recall</i>
<i>Support Vect. Mach.</i>	64%	61%	56%	8%	73%	87%
<i>K-Nearest Neighb.</i>	64%	55%	39%	8%	72%	88%
<i>Random Forest</i>	74%	63%	50%	4%	74%	92%
<i>Neuraal Netwerk</i>	55%	67%	58%	7%	75%	80%

**Tabel 4: Accuracy op textdata voor verschillende modellen (H12.2)**

	Tweets 2021	Tweets 2022	Tekst Metadata	Gecombineerd
SVM	0.840	0.764	0.725	0.803
LSTM	0.790	0.698	0.696	0.738
PAC	0.789	0.732	0.700	0.792
BERT	<b>n.v.t.</b>	0.814	0.720	0.836

**Tabel 5: Accuracy en trainingstijd van concatenate netwerken (H13.2)**

Naam/soort	Train accuraatheid	Test accuraatheid	Tijd
Metadata + userinfo (username + user description)	<b>81.2%</b>	<b>78.9%</b>	<b>994 secondes</b> / 16 minuten
Metadata + tweet	<b>79.6%</b>	<b>77.2%</b>	<b>979 secondes</b> / 16 minuten
Metadata + userinfo + tweet	<b>79.9%</b>	<b>77.6%</b>	<b>1924 secondes</b> / 32 minuten

**Tabel 6: Accuracy en trainingstijd van concatenate netwerken (H15.4.1)**

Indeling (query-key-value)			Train accuraatheid	Test accuraatheid	Tijd (secondes)
Query	Key	Value			
userinfo	<b>n.v.t</b>	tweet	<b>95.2%</b>	<b>74.6%</b>	24.08s
metadata	userinfo	tweet	<b>95.0%</b>	<b>74.3%</b>	28s
metadata	<b>n.v.t</b>	userinfo	<b>90.5%</b>	<b>67.6%</b>	24.21s
userinfo	<b>n.v.t</b>	userinfo	<b>90.3%</b>	<b>67.3%</b>	24.10s
metadata	<b>n.v.t</b>	tweet	<b>91.6%</b>	<b>66.2%</b>	24.12s
tweet	<b>n.v.t</b>	tweet	<b>92.9%</b>	<b>65.3%</b>	24.36s
metadata	<b>n.v.t</b>	metadata	<b>60.9%</b>	<b>61.0%</b>	25.38s

## 20.3 Bijlage C Attention netwerk

