



UHASSELT



Maastricht University

KNOWLEDGE IN ACTION

Faculteit Wetenschappen **School voor Informatietechnologie**

master in de informatica

Masterthesis

Movie time : een user centered groepsaanbevelingssysteem voor films

Mathies Gielen

Scriptie ingediend tot het behalen van de graad van master in de informatica

PROMOTOR :

Prof. dr. Kris LUYTEN

De transnationale Universiteit Limburg is een uniek samenwerkingsverband van twee universiteiten in twee landen: de Universiteit Hasselt en Maastricht University.



UHASSELT

KNOWLEDGE IN ACTION

www.uhasselt.be

Universiteit Hasselt
Campus Hasselt:
Martelarenlaan 42 | 3500 Hasselt
Campus Diepenbeek:
Agoralaan Gebouw D | 3590 Diepenbeek

2020
2021



UHASSELT

KNOWLEDGE IN ACTION



Maastricht University

Faculteit Wetenschappen

School voor Informatietechnologie

master in de informatica

Masterthesis

Movie time : een user centered groepsaanbevelingssysteem voor films

Mathies Gielen

Scriptie ingediend tot het behalen van de graad van master in de informatica

PROMOTOR :

Prof. dr. Kris LUYTEN

Dankwoord

Tijdens het afgelopen jaar heb ik onderzoek gedaan naar een aanbevelingssysteem voor films waarbij de gebruiker centraal staat, genaamd Movie Time. Ik heb enorm veel geleerd in de afgelopen maanden die bijdragen tot mijn verdere carrière als IT-er. Ik wil graag de personen bedanken die mij tijdens dit onderzoek hebben geholpen, hebben gesteund en wijze raad hebben gegeven.

Ten eerste wil ik graag mijn promotor en begeleider Prof. dr. Kris Luyten bedanken die het opzet van mijn thesis heeft ondersteund. Gedurende het hele jaar kon ik rekenen op zijn eerlijke feedback. Ook als ik de bomen even niet meer door het bos zag was hij de persoon die mij naar de volgende stap hielp.

Ook wil ik graag mijn collega's bij Headr, Lennert en Ward, bedanken. Het was een druk jaar met ups en downs. Zij hebben mij steeds gemotiveerd en gesteund tijdens het schrijven van deze thesis. .

Tot slot wil ik mijn vriendin Pauline, mijn ouders en al mijn vrienden bedanken voor hun emotionele steun en ondersteunende gesprekken. Door mijn bevindingen en tegenslagen met hun te delen, heb ik mijn gedachten beter kunnen ordenen en ben ik mijn motivatie blijven behouden. Ik ben hun hiervoor zeer dankbaar.

Samenvatting

Deze thesis omvat de ontwikkeling van een user centered groepsaanbevelingssysteem voor films genaamd Movie Time. Het Movie Time systeem bestaat uit twee grote delen, het opbouwen van een persoonlijk filmprofiel en het vinden van groepsvoorspellingen o.b.v. deze filmprofielen.

Het opbouwen van het filmprofiel gebeurt op een user centered manier waarbij de gebruiker al de controle over het systeem heeft. Voor het opbouwen van de gebruikersprofielen wordt er gebruik gemaakt van tag genomen uit de Movielens dataset. De filmsmaak van een gebruiker is een combinatie van tags uit deze dataset. De tag-dilemma's zorgen ervoor dat de gebruiker op een eenvoudige manier de tags vindt die bij zijn filmsmaak aansluiten.

Het tweede deel van deze thesis betreft het vinden een groepsvoorspelling. Voor de groepsvoorspelling wordt er eerst gekeken of er een doorsneden is in de aanbevolen films van al de afzonderlijke gebruikers. Indien er een dergelijke doorsneden is, is dit de groepsaanbeveling. Wanneer deze doorsnede leeg is, gaat het systeem op zoek naar de common-ground. De common-ground is een aantal tags die de gemeenschappelijke smaak van de groep voorstelt. Net zoals bij een individuele gebruiker kan deze gemeenschappelijke smaak dan gebruikt worden om films te voorspellen.

Het doel van deze thesis is een formatieve evaluatie en het verzamelen van feedback over het Movie Time platform. Het doel van de evaluatie is afoetsen of de voorgestelde aanpak werkt en gerichte feedback verzamelen voor de implementatie van een volgende versie van dit platform. Voor deze evaluatie werd de proof of concept online gezet (Movietime.be) en werd er bij zestien test gebruikers feedback verzameld door middel van een vragenlijst. De feedback van de test gebruikers was overwegend positief en er werd belangrijke feedback verzameld voor de ontwikkeling van een volgende versie.

Inhoudsopgave

Inhoudsopgave	7
I Inleiding	9
1 Introductie en motivatie	11
1.1 Scenario	12
2 Gerelateerd werk	15
2.1 Aanbeveling	15
2.2 Groepsaanbeveling	16
2.3 Tag genomen	17
2.4 Serendipiteit	18
II Methode	19
3 Movie Time implementatie	21
3.1 Algemene werking	21
3.2 Data	23
3.3 Pre processing	24
3.4 Applicatie	24
4 Aanbevelingsalgoritmen	27
4.1 Film aanbeveling	28
4.2 Tag aanbeveling	28
5 Persoonlijk profiel	31
5.1 Genre game	31
5.2 Tag-dilemma's	33
5.2.1 Implementatie	33
5.3 Film selectie	35
5.3.1 Implementatie	35

6 Groepsvoorspelling	37
6.1 Filtering	37
6.2 Common-ground	37
III Validatie en discussie	39
7 Onderzoeksmethode	41
7.1 Deelnemers	41
8 Analyse en Resultaten	43
8.1 Kijkgedrag individueel	43
8.2 Kijkgedrag in groep	43
8.3 Persoonlijk profiel	44
8.4 Groepsvoorspelling	45
8.5 Algemeen	46
9 Discussie	47
9.1 Filmprofiel	47
9.2 Groepsvoorspelling	48
9.2.1 Voorspelling op basis van films	48
9.2.2 Voorspelling op basis van tags	49
9.2.3 Common-ground	49
9.2.4 Vergelijking	50
9.3 Algemene discussie	50
10 Toekomstig werk	53
11 Conclusies	55
A Vragenlijst	63

Deel I
Inleiding

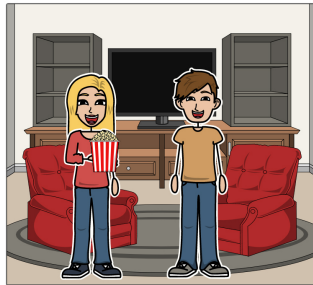
Hoofdstuk 1

Introductie en motivatie

De tijd dat je enkel de films kon kijken die er op dat moment uitgezonden werden is verleden tijd. Tegenwoordig zijn alle films en tv-series binnen handbereik. Deze toename aan beschikbare content heeft belang van persoonlijke aanbevelingen doen toenemen. Streamingsdiensten zoals Netflix hebben hun populariteit dan ook voor een groot deel te danken aan een goed aanbevelingsalgoritme. Hoe beter de aanbeveling is, hoe langer de gebruikers blijven kijken en dus ook blijven betalen. Zoeken naar een serie of film die aansluit bij jou interesses is vandaag de dag dan ook niet moeilijk. Het wordt echter veel ingewikkelder wanneer je in groep een geschikte film wilt kiezen. Iedereen heeft zijn eigen smaak, maar die smaak kan veranderen afhankelijk van verschillende elementen, zoals het gezelschap, je humeur,...

Het doel van deze thesis is dan ook een proof of concept te ontwikkelen van een platform dat het keuzeprocess in groepen vereenvoudigt. De uitdaging in deze thesis is tweeledig. Ten eerste moet het systeem in staat zijn om een accurate voorstelling van een gebruiker zijn filmsmaak te maken zonder dat er voorkennis of kijk geschiedenis van deze gebruiker aanwezig is. De tweede uitdaging bestaat uit een aanbeveling te genereren die bij al deze leden aansluit. Het doel is een aanbevelingsalgoritme te ontwikkelen waar de gebruiker centraal staat. In de praktijk werken aanbevelingsalgoritmen vaak op data die verzameld worden door een gebruiker te volgen en zijn gedrag vast te leggen. Vervolgens krijgt de gebruiker voorspellingen op basis van data waar hij geen controle over heeft en vaak niet van het bestaan af weet. De drijfveer achter persoonlijk aanbevelingen is vaak commercieel. In de paper van Gomez-Uribe et al [5] wordt beschreven hoe het Netflix algoritme gebruik maakt van tientallen parameters die ze verzamelen van gebruikers. Ook lichten ze toe dat dit aanbevelingsalgoritme Netflix jaarlijks 1 miljard dollar opbrengt. Onze aanpak verschilt hiervan door de gebruiker centraal te plaatsen en hem alle controle over zijn gebruikersprofiel te geven. De extra uitdaging hierbij is een methode ontwikkelen zodat de gebruiker eenvoudig zijn voorkeuren en filmsmaak kan opgeven.

1.1 Scenario



Fien en Steven gaan samen een film kijken.



Ze scrollen door het aanbod van Netflix op zoek naar een film die hun beiden aanspreekt.



Na een lange zoektocht worden ze dit beu en kiezen ze een film die Steven al ooit gezien heeft.

Figuur 1.1: Scenario wanneer er geen gebruik gemaakt wordt van het systeem

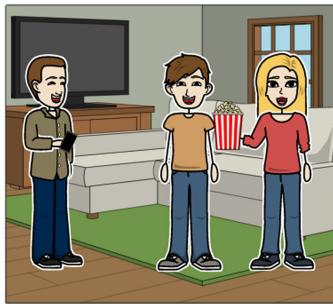
Op Figuur 1.1 wordt het scenario getoond van een typische filmavond. Fien en Steven spreken af om samen een film te kijken. Ze beginnen door het aanbod van Netflix te scrollen. Ze kunnen filteren op een genre en krijgen vervolgens een lijst met covers en titels te zien. De ordening van deze films gebeurt op basis van de filmhistoriek van de eigenaar van het account. Dit scrollen door een eindeloos aanbod aan films wordt al snel frustrerend en er wordt uiteindelijk een film gekozen die niet de volledige voorkeur van Fien of Steven krijgt.

Figuur 1.2 visualiseert een interactieve manier om een filmkeuze te maken. Louis, Steven en Fien spreken af om een film te kijken. Het is de eerste keer dat Louis het duo vergezelt. Hij gebruikt zijn GSM om een account te maken op het systeem. Steven en Fien maakten al eerder gebruik van Movie Time en zij loggen in op hun bestaand account. Steven maakt een nieuwe groep aan. Vervolgens worden de vrienden lid van deze groep door de code in te voeren. Iedereen kan nu afzonderlijk zijn film-profiel vervolledigen door dilemma's te beantwoorden. Op basis van deze verkregen data achterhaalt het systeem de filmsmaak van elke deelnemer.

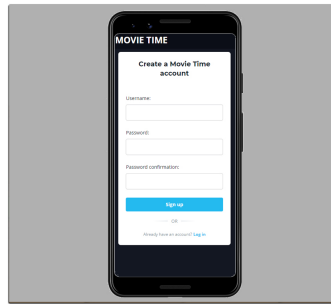
Wanneer Steven het laatste dilemma heeft beantwoord, komt hij terug op de groepspagina terecht. Hier ziet hij het resultaat van de vragenlijst die hij zojuist voltooid heeft. Dit resultaat is een lijst van films en een lijst van tags die een abstractie vormen van zijn filmsmaak.

Wanneer hij op de groepsvoorspelling pagina kijkt, ziet hij de boodschap dat het film-profiel van Louis en Fien nog niet compleet is. Steven moet dus nog even wachten tot zij hun vragenlijst ingevuld hebben.

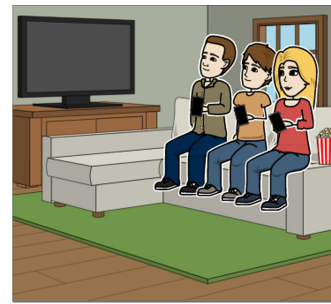
Wanneer al de leden van de groep klaar zijn, verschijnt er een knop om de groepsvoorspelling te berekenen. Bij het berekenen van de groepsvoorspelling zijn er twee mogelijke scenario's. In het eerste geval is er een overlapping tussen de films die al de groepsleden



Louis, Steven en Fien gaan samen een film kijken.



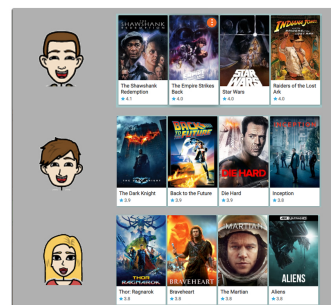
Louis gebruikt het systeem voor het eerst en registreert zich bij Movie Time



Ze maken een nieuwe groep aan en worden allemaal lid van deze groep.



Al de leden vervolledigen hun filmprofiel d.m.v. een spelletje.



Een algoritme berekent voor iedere gebruiker aanbevolen films

Figuur 1.2: Scenario wanneer er gebruik gemaakt wordt van het systeem deel 1

na het voltooien van de vragenlijst voorspeld kregen. In dit geval gaan we verder met het scenario in figuur 1.3.

De gebruikers krijgen de films in de doorsnede te zien, dit zijn de films die aansluiten bij al de leden van de groep.

Indien de doorsnede tussen de voorspelde films leeg is, wordt er verdergegaan met het scenario in figuur 1.4. Omdat er geen gemeenschappelijke films gevonden werden voor al de gebruikers gaan we verder met het zoeken van een common-ground tussen de gebruikers. Deze common-ground zijn gemeenschappelijke elementen die voorkomen in de filmsmaak van al de gebruikers. De common-ground wordt in het systeem voorgesteld als een pseudo gebruiker die als filmsmaak de gemeenschappelijke elementen van al de gebruikers heeft. Wanneer de common-ground gevonden is, gaat het systeem op zoek naar films die aan deze common-ground voldoen. Deze films worden vervolgens aan de gebruikers voorgesteld. Nu kan de filmavond beginnen!

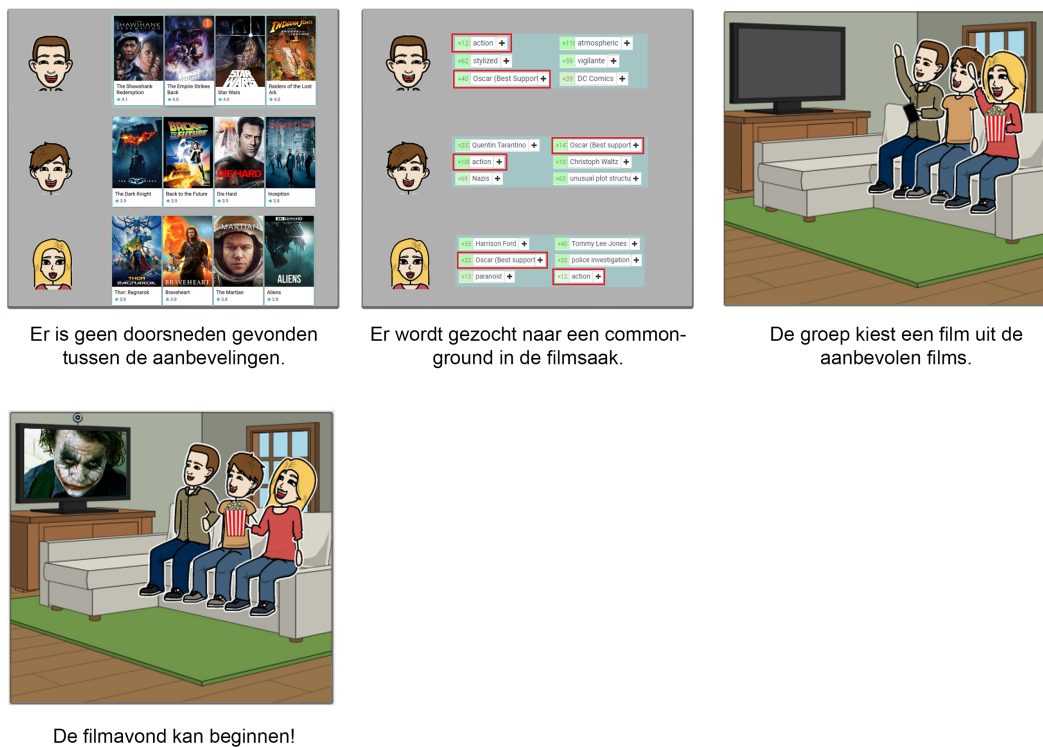


Er wordt een doorsneden gevonden tussen de aanbevelingen.

De groep kiest een film uit de aanbevolen films.

De filmavond kan beginnen!

Figuur 1.3: Deel 2A - Gevolg wanneer er een doorsneden gevonden wordt



Er is geen doorsneden gevonden tussen de aanbevelingen.

Er wordt gezocht naar een common-ground in de filmsaak.

De groep kiest een film uit de aanbevolen films.

De filmavond kan beginnen!

Figuur 1.4: Deel 2B - Gevolg wanneer er geen doorsneden gevonden wordt

Hoofdstuk 2

Gerelateerd werk

2.1 Aanbeveling

Er zijn veel verschillende types van aanbevelingssystemen met elk hun voor en nadelen. In de paper van Debashis Das et al [3] wordt er een overzicht gegeven van de verschillende types van algoritmen. Een content-based filtering algoritmen sluit het best aan bij onze noden. In de paper van Poonam B.Thorat et al [1] bespreken ze de voordelen van een content-based filtering systeem. Een content-based filtering systeem zorgt voor een transparantie naar de gebruiker. Ook heeft dit systeem minder last van het cold-start probleem, wat wilt zeggen dat er initieel niet voldoende informatie in het systeem aanwezig is om een goede voorspelling te doen. Dit probleem wordt besproken in het onderzoek van Xuan Nhat Lam et al [9]. Dit zijn twee belangrijke voordelen voor het Movie Time platform aangezien er zoveel mogelijk transparantie voor de gebruikers wordt voorzien en er gewerkt wordt met gebruikers waar nog geen gegevens van beschikbaar zijn.

In het boek van Michael J. Pazzani en Daniel Billsus [15] wordt beschreven hoe een content-based aanbevelingssysteem opgebouwd moet worden. Zij rapporteren een methode om de inhoud van de films te beschrijven, een methode om een gebruikersprofiel op te bouwen zodat het beschrijft welke elementen de gebruiker graag heeft en een methode om beide elementen met elkaar te vergelijken. In ons systeem worden tags gebruikt om de films te beschrijven en gebruiken we dezelfde tags om een gebruikersprofiel op te bouwen. Hoofdstuk 4 omschrijft hoe beide elementen onderling vergeleken worden. De paper van Hui Li [11] introduceert een content-based aanbevelingssysteem voor films. Ze doen dit door data te minen uit het social media profiel van de gebruiker. Hiermee wordt het cold-start probleem voorkomen en kan de gebruiker zonder enige input een voorspelling op maat krijgen. Dit verschilt van de user-centered aanpak die in deze thesis toegelicht wordt omdat in dit onderzoek de gebruiker al de controle krijgt.

2.2 Groepsaanbeveling

Door de snelle ontwikkeling van de informatie technologie sector zijn aanbevelingsalgoritmen een als maar belangrijkere tool geworden voor mensen om de nodige info te vinden in de overdaad van beschikbare data. De meeste aanbevelingsalgoritmen zijn echter ontworpen om een aanbeveling te doen aan een individuele gebruiker. Binnen deze thesis wordt een manier onderzocht om aanbevelingen te doen voor een groep van mensen. In deze sectie in deze sectie wordt er dieper ingegaan op bestaande aanpakken om groepsaanbevelingen te doen.

In de paper van Dina Goren-Bar et al [6] stellen ze een systeem voor dat tv-programma's voorspelt op basis van de gezinsleden die aan het kijken zijn. Bij dit systeem wordt er ook rekening mee gehouden dat de voorkeur van deelnemers afhankelijk is van het gezelschap. Om te beginnen worden al de familieleden als een stereotype gebruiker voorgesteld samen met de momenten dat deze gebruiker meestal naar tv kijkt. Het systeem monitort dan de effectieve kijk keuzes van de gebruikers en vergelijkt dit met de voorspelling die hij zelf doet op basis van de profielen. Deze vergelijking gebruikt hij vervolgens om de gebruikersprofielen van de kijkers te updaten. Het voordeel van een dergelijk systeem is dat hiervoor weinig initiële info van de gebruiker nodig is en het systeem al doende beter en beter wordt. In onze toepassing werkt deze aanpak echter niet. Het bekijken van een film in groep is iets wat meer tijd vraagt en minder frequent voorkomt dan het bekijken van een tv-programma. Indien we onze keuzes enkel zouden laten beïnvloeden door de vorige films die je gekeken hebt met diezelfde groep zou iedere groep eerst een aantal films moeten kijken alvorens er gepersonaliseerd aanbevelingen beschikbaar zijn. Dit is wel een aanpak die in de toekomst bij Movie Time gebruikt kan worden om de initiële aanbeveling over de tijd te verbeteren.

In de paper van De Pessemer et al [17] evalueren ze groepsaanbevelingen op basis van het samenvoegen van de voorkeuren van de verschillende gebruikers. Dit samenvoegen doen ze op twee verschillende methoden. In de eerste strategie gaan ze de individuele aanbevelingen van al de afzonderlijke groepsleden combineren tot een groepsaanbeveling. In de tweede strategie groeperen ze de voorkeuren van al de gebruikers tot groepsvoorkeuren om dan vervolgens de groepsvoorkeuren te behandelen als een pseudo gebruiker en met een traditioneel aanbevelingsalgoritme een aanbeveling te doen. De aanpak die wij toepassen in het Movie Time systeem kan gezien worden als een hybride van beide strategieën. In eerste instantie wordt er gezocht naar een overlap in de individuele aanbevelingen. Indien deze niet gevonden wordt, worden de voorkeuren van al de gebruikers samengevoegd en wordt er op basis van deze samengevoegde voorkeuren gezocht naar groepsaanbevelingen. De manier waarop dit gebeurt verschilt met de aanpak bij Movie Time. Bij de bovenstaande methoden worden de voorkeuren van de gebruikers voorgesteld als ratings, iets wat voor ons systeem niet optimaal is. Er zijn twee belangrijke redenen waarom er bij Movie Time geen gebruik gemaakt wordt van ratings als voorstelling van een filmsmaak. Ten eerste zijn ratings niet flexibel genoeg om zich aan te passen

aan een stemming of gezelschap. Het andere probleem noemt het cold-start probleem. Wanneer we een voorspelling willen doen op basis van ratings moeten er eerst voldoende ratings van iedere persoon beschikbaar zijn. In het onderzoek van Sang-Min Choi et al [2] wordt dit probleem omzeild door de voorspelling te doen op basis van de genres van een film in plaats van beoordelingen. Aan iedere film zijn een of meerdere genres toegewezen. Deze toewijzing wordt gedaan door de makers van de film zelf. We kunnen er dus van uit gaan dat deze data kwalitatief is. Dit systeem berekent eerst de correlatie tussen al de verschillende genres van de verschillende films. Het systeem gaat vervolgens een aanbevelingsscore berekenen op basis van de gemiddelde score van de film, de favoriete genres van de gebruiker en de genre-correlatie informatie. Deze aanpak sluit dicht aan bij Movie Time. Enkel gebruik maken van genres zorgt er echter voor dat er slechts een heel beperkt smaakprofiel opgebouwd kan worden. Daarom wordt er in dit onderzoek zowel gebruik gemaakt van genres als van inhoudelijke tags.

2.3 Tag genomen

In de paper van Jesse Vig et al [22] introduceren ze het concept tag genome. In tegenstelling tot het traditionele tag model waar tags binair worden voorgesteld (een item heeft de tag of heeft de tag niet) geeft de tag genome aan hoe relevant een tag is voor een bepaald item. Dit principe passen ze vervolgens toe in Movie Tuner. Movie Tuner is een systeem dat de gebruiker interactief op zoek laat gaan naar films door de films onderling te vergelijken op basis van tag genomen.

In het artikel van Harper et al [7] wordt beschreven hoe het tag genome geïmplementeerd werd in de Movielens dataset. Dit maakt de Movielens dataset de ideale dataset voor de implementatie van Movie Time. De gebruikte data wordt verder toegelicht in sectie 3.2.

Er zijn al verschillende onderzoeken die ons voor gaan in de implementatie van een aanbevelingssysteem op basis van tag genome. In de paper van Syed Mohd Ali et al [14] onderzoeken ze een methode om content gebaseerde aanbevelingen voor films te doen op basis van de tag genomes aanwezig in de Movielens dataset. In de paper van Maria S. Pera en Yiu-Kai Ng [16] wordt het groepsaanbevelingssysteem GroupReM voorgesteld, net zoals Movie Time werkt GroupReM ook op de Movielens dataset en maakt hij gebruik van de genome tags. Het grote verschil is dat GroupReM werkt op basis van de films en tags op de Movielens profielen. Onze aanpak moet een gebruiksvriendelijk en interactief systeem worden waar gebruikers in enkele minuten hun filmsmaak kunnen ingeven of wijzigen zonder dat ze hierbij actief op zoek moeten naar de films die het best aansluiten bij hun filmsmaak.

2.4 Serendipiteit

Aangezien het bij een groepsaanbevelingssysteem niet altijd mogelijk is om de meest accurate aanbeveling voor iedere persoon te vinden is serendipiteit een belangrijke metriek. Sean B Seymore [19] beschrijft het concept serendipiteit als volgt: “Het proces om iets van waarde te vinden dat aanvankelijk niet gezocht was” In de paper van Guy Shani et al [20] over de evaluatie van aanbevelingssystemen beschrijven ze serendipiteit als een maat staaf voor hoe verrassend een succesvolle aanbeveling is.

In de paper van Yuan Cao Zhang et al [23] wordt een ideaal aanbevelingsalgoritme omschreven als volgt: “Een ideaal aanbevelingssysteem zou de acties van een vertrouwde vriend of expert moeten nabootsen en een gepersonaliseerde verzameling aanbevelingen moeten produceren die een evenwicht vinden tussen de gewenste doelen van nauwkeurigheid, diversiteit, nieuwheid en serendipiteit.” Buiten de gekende parameters voor een aanbevelingsalgoritme is serendipiteit dus ook een belangrijk element voor een aanbevelingssysteem. Een aanbeveling die op het eerste zicht niet lijkt aan te sluiten bij een gebruiker zijn smaak kan na het kijken toch als goed ervaren worden. Bij de finale evaluatie is het dus belangrijk om hier rekening mee te houden.

Deel II
Methode

Hoofdstuk 3

Movie Time implementatie

In dit deel wordt de methode, de werking en de algemene aanpak van het Movie Time platform besproken. Ook zal er een gedetailleerde uitleg gegeven worden bij de verschillende geïmplementeerde algoritmen.

3.1 Algemene werking

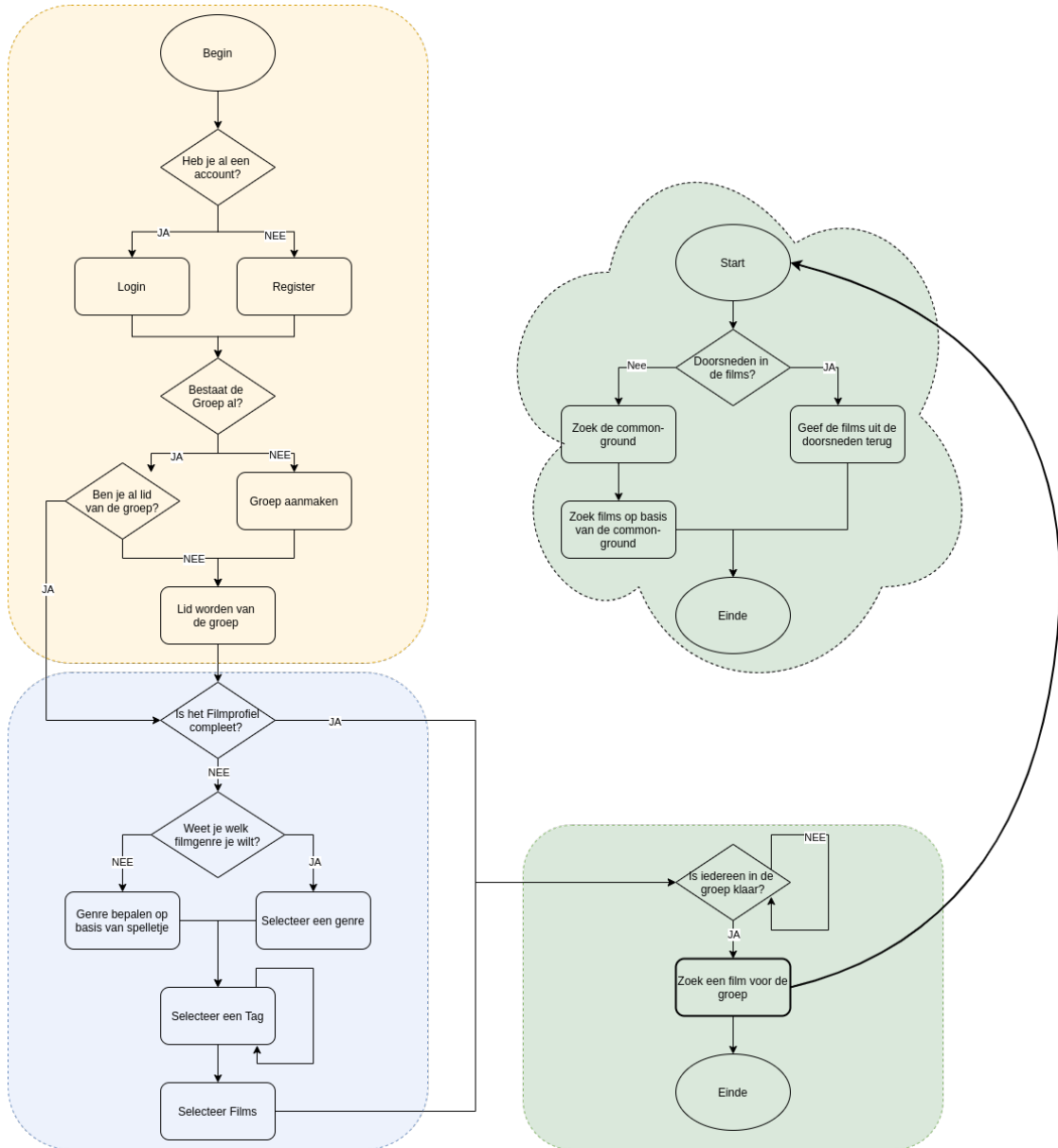
De werking van het systeem wordt in schema 3.1 gevisualiseerd met de verschillende stappen die het systeem doorloopt.

De eerste stap is het aanmaken van een account. Indien je het platform al eerder bezocht hebt, kan je inloggen met een bestaand account. Na het inloggen krijg je een overzicht van al de groepen waar je lid van bent. Een nieuwe groep aanmaken gebeurt door de groepsnaam in te geven en vervolgens de invite-code aan de andere groepsleden te bezorgen. Deze personen kunnen vervolgens de invite-code in het daar voor voorziene veld invullen en zo lid worden van de groep.

Vervolgens vervolledigen al de leden hun persoonlijk filmprofiel. Voor iedere groep kan er een ander persoonlijk filmprofiel samengesteld worden.

Het samenstellen van het persoonlijk filmprofiel kan op twee manieren. Indien de deelnemer niet weet welk filmgenre hij wil kijken, wordt het filmgenre achterhaald door middel van een klein spel. Wanneer het filmgenre bepaald is, wordt het filmprofiel verder gespecificeerd aan de hand van tag-dilemma's. Bij deze dilemma's is het de bedoeling dat de gebruiker uit een reeks van tags telkens de tag kiest die het dichtst bij zijn filmsmaak aansluit. Op deze manier worden de films uit de dataset inhoudelijk gefilterd. Enkel de films die dicht aansluiten bij de filmsmaak van de gebruiker worden getoond. De gebruiker kiest hieruit de films die hem het meest aanspreken. Nu is het persoonlijk filmprofiel compleet.

De groepsvoorspelling kan starten wanneer al de leden hun persoonlijk profiel voltooid hebben. Het doel van deze groepsvoorspelling is het vinden van een film die past bij de smaak van al de leden. Het systeem controleert eerst of er een doorsneden is in de aanbevolen films van iedere gebruiker. De groepsvoorspelling is voltooid indien een dergelijke doorsneden gevonden wordt. In de meeste gevallen zal een dergelijke doorsneden

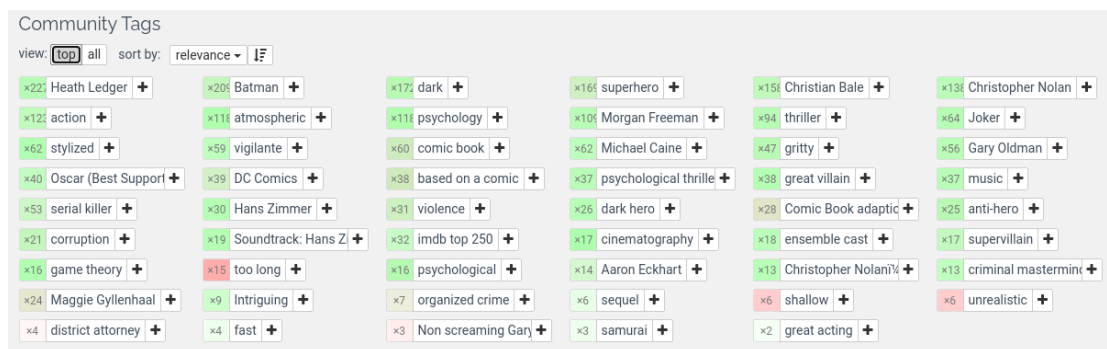


Figuur 3.1: Schematische voorstelling van het platform

echter leeg zijn. Dan moeten we op zoek naar de common-ground van de groep. Deze common-ground is een combinatie van tags die de gebruikers gemeenschappelijk hebben. Aangezien elke film vele tags bevat is het vinden van een doorsneden in deze tags veel waarschijnlijker. Deze gevonden common-ground wordt dan gebruikt om films aan te bevelen aan de groep.

3.2 Data

Zoals besproken in sectie 2.1 werken veel aanbevelingsalgoritmen op basis van het kijkgedrag van andere gebruikers. Men gaat dan in een grote dataset met reviews op zoek naar matchende gebruikers. Deze manier van aanbevelen heeft geen weet van de inhoud van een film of wat een gebruiker juist aanspreekt in films. Binnen dit onderzoek staat de gebruiker centraal en is het daarbij essentieel dat hij controle heeft over zijn filmsmaak. Hiervoor is een dataset nodig met voldoende inhoudelijke info over zo veel mogelijk films. De Movielens dataset [7] bevat niet enkel de standaard informatie zoals titel, jaar, genres... maar bevat ook voor iedere film een groot aantal tags. Deze tags geven een inhoudelijke abstractie van een film. In figuur 3.2 zijn de tags te zien



Figuur 3.2: Movielens [7] tags voor de film 'The dark knight'

voor de film 'The dark knight'. Er wordt voor iedere tag aangegeven hoeveel personen deze tag bevestigen. Deze waarde geeft de relevantie van de tag weer. Zoals te zien is in het voorbeeld kunnen tags inhoudelijke elementen van de film bevatten zoals 'serial killer', 'super hero'... Ook informatie over de cast, regisseur en gewonnen prijzen kan beschreven worden als tag. De combinatie van al deze tags vormen een soort van unieke vingerafdruk die al de belangrijke eigenschappen van die film omvat.

Deze tags zijn dus de perfecte manier om films inhoudelijk te filteren. Movielens [7] stelt verschillende versies van de dataset ter beschikking, binnen dit onderzoek wordt er gebruik gemaakt van de 25M Dataset. Deze dataset bevat 25 miljoen film beoordelingen en een miljoen tags die door 162,000 gebruikers zijn toegewezen aan 62000 films. De data is vrijgegeven in december 2019, waardoor de meest recente films in deze dataset gemaakt zijn in 2019. Om de dataset verder aan te vullen maken we gebruik van info die te vinden is op IMDB [13]. Zo zal de cover-foto die in het systeem getoond wordt

opgehaald worden van op IMDB[13].

3.3 Pre processing

Aangezien een grote dataset van 62000 films en 1 miljoen tags te zwaar is om in een realtime web-applicatie te verwerken, is een correcte preprocessing van de data belangrijk. Er werd gekozen voor de methode die Johnson Kuan beschreef in het volgende artikel: “How to Build a Simple Movie Recommender System with Tags” [8]

De eerste stap is het inladen van de films, genome scores en de genome tags datasets in een Pandas dataframe[12]. Het film dataframe bevat nu de kolommen movieId, title en tags. De entries in de titel kolom zien er als volgt uit: “NeverEnding Story III, The (1994)”. Om deze data verder te kunnen gebruiken moet de titel kolom dus eerst opgeruimd worden. Het release-jaar wordt aan een aparte kolom toegevoegd. Voorzetsels zoals The, An en A staan in de dataset achter de effectieve titel. Om de titel gemakkelijker leesbaar te maken wordt het voorzetsel voor de titel geplaatst. Het film dataframe is nu klaar voor gebruik. Vervolgens wordt het tag dataframe gepreprocessed. Omdat niet al de tags even relevant zijn, wordt voor iedere film enkel de honderd meest relevante tags gebruikt.

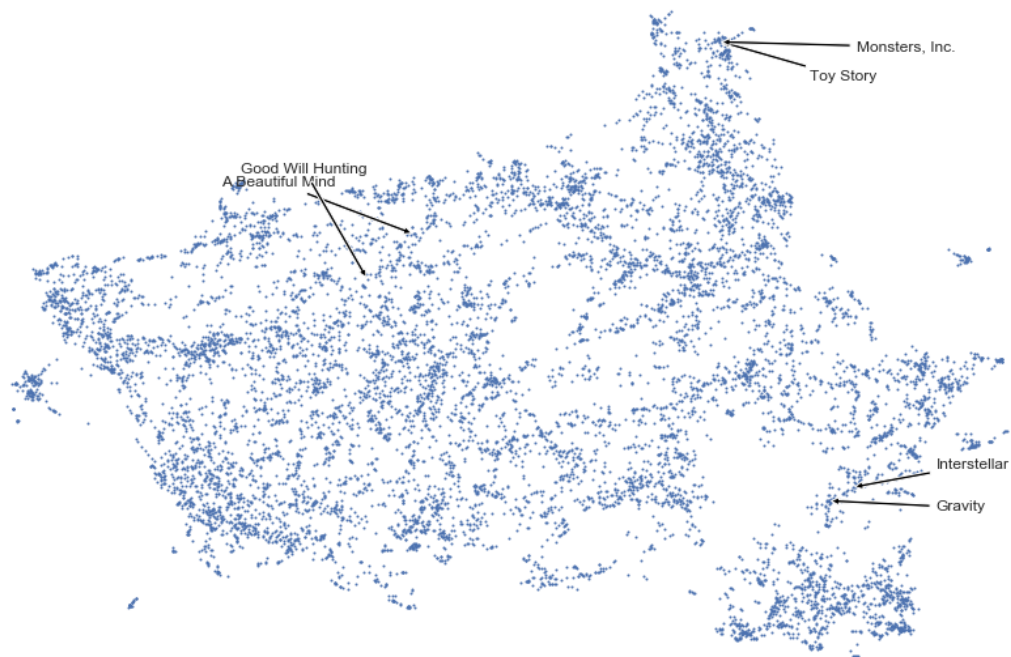
Zoals voorgesteld in de paper van Quoc Le en Tomas Mikolov [10] wordt de Doc2Vec techniek gebruikt. Deze techniek leert een mapping tussen een film (document) en een K dimensionale vector. Bij deze mapping krijgen vectors van films met veel gemeenschappelijke kenmerken een kleine onderlinge afstand. Hiervoor wordt er gebruik gemaakt van de Doc2Vec functie in het Gensim-python framework [18]. Na het trainen van dit model wordt er voor iedere film een vector berekend die opgeslagen wordt in het systeem. Deze vectoren worden in verdere stappen gebruikt om aanbevelingen te doen.

Een grafische voorstelling van al de films uit de Movielens dataset en hun overeenkomstige film-vector is te zien in afbeelding 3.3. Hier is nog eens duidelijk te zien dat gelijkaardige films zoals “Monsters inc”. en “Toy Story” gemapt worden op punten heel kort bij elkaar. Films die weinig gemeenschappelijke kenmerken bevatten zoals “Toy Story” en “Interstellar” hebben dan weer een grotere onderlinge afstand.

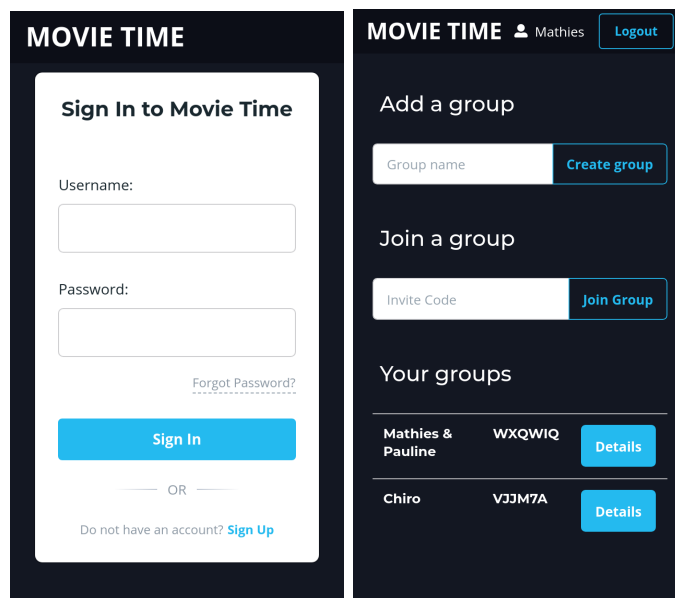
De laatste stap is het opslaan van het Pandas [12] dataframe op de server. Wanneer de server herstart wordt, kunnen het model en de data-frames ingeladen worden. Hierdoor hoeft het trainen en mappen niet opnieuw te gebeuren. Dit zorgt ervoor dat het herstarten slechts enkele seconden duurt in plaats van enkele minuten.

3.4 Applicatie

In deze sectie wordt het gele blok in figuur 3.1 toegelicht. Dit is de interactie van de gebruiker met de applicatie. Bij het ontwerp/implementatie van de applicatie is er met een aantal belangrijke elementen rekening gehouden. De applicatie zal in de meeste gevallen mobiel gebruikt worden. Een vlotte en intuïtieve werking van de applicatie op mobiele apparaten is daarom belangrijk. Indien de leercurve te hoog is, is de kans groter dat gebruikers zouden afhaken bij het gebruik van de applicatie.

Movie Vector (Doc2Vec Output Visualized in 2D with UMAP)**Figuur 3.3:** Visualisatie van MovieLens films in 2D [8]

Om de applicatie zo gebruiksvriendelijk mogelijk te maken op mobiele apparaten zijn we vanuit een mobile-first aanpak vertrokken bij het maken van de layout. Hierdoor is al de tekst en andere elementen duidelijk zichtbaar op het kleine scherm van een gsm. De verwerking van de data en de implementatie van de web-applicatie gebeurt in Python. Het gebruikte Python framework is Django in combinatie met een PostgreSQL database. Django zorgt ervoor dat het als ontwikkelaar makkelijker is om snelle en veilige web applicaties te maken.



Figuur 3.4: Screenshots van de applicatie

Hoofdstuk 4

Aanbevelingsalgoritmen

Er bestaan twee grote categorieën van aanbevelingsalgoritmen, collaborative filtering of content-based filtering. Een collaborative filtering algoritme werkt volledig op basis van het gedrag van de gebruiker in het verleden. Meer specifiek worden er op basis van de smaak en keuzes in het verleden andere gebruikers gezocht met een gelijkaardige smaak. Op basis van die gelijkaardige profielen worden er dan voorspellingen gedaan aan de gebruiker

Een andere methode voor het vinden van aanbevelingen is content-based filtering. Bij content-based filtering wordt er gebruik gemaakt van de inhoud van films die je leuk vindt om andere films met gelijkaardige kenmerken te kunnen vinden.

Er zijn twee problemen bij het gebruik van een Collaborative filtering algoritme in ons systeem. Collaborative filtering werkt enkel goed wanneer er veel data beschikbaar is over de kijkgeschiedenis van de gebruiker. Ten tweede kan de Collaborative filtering niet goed anticiperen met verandering van smaak van de kijker. Aangezien er binnen dit onderzoek een voorspelling gedaan wordt op basis van minimale info over de gebruikers én rekening houdend met de verandering van de filmsmaak op basis van de mede-kijkers, kan de collaborative filtering niet gebruikt worden. De twee voorgaande nadelen zijn niet van toepassing bij content-based filtering. Om deze reden wordt voor dit systeem de implementatie van een content-based filtering algoritme gehanteerd. Deze content-based filtering gaan we doen op basis van de filmtags uit de Movielens dataset.

Zowel voor het opbouwen van het persoonlijk profiel als bij het zoeken van een groepsvoorspelling wordt er gebruik gemaakt van twee verschillende aanbevelingsalgoritmen. Alvorens we de toepassing van deze algoritmen toelichten in sectie 5 en sectie 6 wordt er hier de implementatie van deze algoritmen toegelicht.

Het eerste algoritme is de Jaccard Index om de gelijkheid tussen twee sets van film-tags te berekenen. Dit algoritme wordt gebruikt om op basis van enkele tags te bepalen welke films hierbij het dichtst aansluiten. Het tweede algoritme is de Cosine Similarity van film-vectors. Dit algoritme wordt gebruikt om de dichtst aansluitende films te vinden op basis van een set van films.

4.1 Film aanbeveling

Het eerste algoritme krijgt als input een lijst van films en geeft als output een lijst van films die het meest gelijkend zijn aan de films die als input gegeven werden. De eerste stap in dit algoritme is de berekening van de search-vector. Deze search-vector is het gemiddelde van de film-vectors die als input gegeven werden. Vervolgens wordt deze search-vector als input van de *Doc2Vec.most_similar()* [18] functie gebruikt. Deze methode berekent de cosine similarity tussen deze search-vector en vectors van al de andere films. Als resultaat geeft de methode de n meest gelijkende films terug.

Listing 1 Gelijkaardige films op basis van films

```
def getPrediction(movies, amount):
    #compute user vector as an average of movie vectors in movies
    search_movie_vector = np.zeros(shape=mv_tags_vectors.shape[1])
    for mv in movies:
        mv_index = mv_tags_list[mv_tags_list["movieId"] == mv].index.values[0]
        search_movie_vector += mv_tags_vectors[mv_index]
    search_movie_vector /= len(movies)
    # find movies similar to user vector to generate movie recommendations
    similar_movies = model.docvecs.most_similar(positive=[search_movie_vector],
                                                topn=amount)

    output = []
    for i, j in similar_movies:
        movie_sim = mv_tags_list.loc[int(i), "movieId"]
        if movie_sim not in movies:
            output.append(movie_sim)
    return output
```

4.2 Tag aanbeveling

Het tag aanbeveling algoritme is te zien in listing 2. Dit algoritme krijgt als input een lijst van tags en geeft als output een lijst van films die het dichtst bij deze input tags aansluiten. De functie kan drie input attributen mee krijgen, *tags* dit is een lijst met de tags waar men gelijkende films voor wilt zoeken, *amount* dit is het aantal films dat je wilt terug krijgen. Met het optionele attribuut *movies_list* kan er een lijst van films meegegeven worden waarin het algoritme moet zoeken. Dit laatste wordt bijvoorbeeld gebruikt wanneer er enkel films van een bepaald genre gebruikt mogen worden.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Figuur 4.1: Formule voor het berekenen van de Jaccard Index

Vervolgens wordt voor iedere film de Jaccard index ten opzichte van de input tags berekend. De Jaccard Index geeft de gelijkenis tussen twee sets weer. De formule die hiervoor gebruikt wordt is te zien in figuur 4.1. Deze formule wordt uitgevoerd met als argument de tags waar we een gelijkende film voor willen vinden. Op deze manier wordt er voor al de films een Jaccard index berekend. De film die de de hoogste Jaccard index heeft is bijgevolg de film die het meest lijkt op de set van tags die als input gegeven werden. Door gebruik te maken van de Pandas library [12] in Python kan deze Jaccard index op een efficiënte manier berekend worden voor een dergelijke grote dataset. Als output geeft het algoritme dan de n films met de hoogste Jaccard Index

Listing 2 Gelijkaardige films op basis van tags

```
def tagSimilarity(tags, amount, movies_list=None):
    # use custom list of tags or use all tags
    if not movies_list:
        #No custom list is used get all movies from global
        movies_list = all_movies_list
    # movies_js is al relevant movie info and a jaccard similarity
    movies_js = movies_list[['movieId', 'Title', 'tag_list', 'tag']]
    movies_js['jaccard_sim'] = movies_js.tag_list.map(
        lambda x:
            len(set(x).intersection(set(tags))) /
            len(set(x).union(set(tags)))
    )
    return movies_js.sort_values(by='jaccard_sim',
                                ascending=False)['movieId'].head(amount).values.tolist()
```

Hoofdstuk 5

Persoonlijk profiel

De volgende stap is het opstellen van het persoonlijk profiel, in afbeelding 3.1 is dit deel in het blauw gekleurd. Het opstellen van het persoonlijk profiel is de eerste belangrijke stap richting een goede filmvoorpelling voor de groep. De ideale film voor een groep voldoet aan de verwachtingen van al de individuele leden van de groep. Hiervoor moet het systeem van iedere gebruiker weten wat zijn filmsmaak is. Het definiëren en communiceren van ieder zijn filmsmaak binnen een groep kan ervaren worden als een moeizaam proces.

Het persoonlijk filmprofiel moet de voorkeuren en wensen van elk individu binnen de groep op een gestructureerde manier weergeven om later te kunnen gebruiken om de groepsvoorspelling te doen (cf. sectie 6). Om het opstellen van dit profiel zo gemakkelijk mogelijk te laten verlopen bestaat het uit 2 eenvoudige spelletjes. Het eerste is om het gewenste filmgenre te achterhalen (sectie 5.1), het tweede is vervolgens om de filmsmaak verder inhoudelijk vorm te geven (sectie 5.2). Omdat veel mensen al zeker zijn van het genre dat ze willen kijken is het mogelijk om het eerste spelletje over te slaan en zelf een genre te kiezen.

5.1 Genre game

Wanneer de gebruiker aangeeft nog niet te weten welk genre hij wilt kijken wordt hij meegenomen in een interactief verhaal waar verschillende keuzes gemaakt moeten worden. De vragen voor dit verhaal worden opgehaald uit een JSON-bestand en zijn dus eenvoudig aan te passen. In listing 3 is te zien hoe een vraag in het JSON-bestand wordt voorgesteld. Iedere vraag bestaat uit een “question”, een “type” en “answers”. De type geeft aan om welk vraagtype het gaat. Op dit moment worden er vier types van vragen ondersteund:

- Question: Dit vraag-type is voor de vragen waarbij de antwoorden in volzinnen worden weergegeven. Er kan hierbij 1 antwoord gekozen worden.
- Range: Bij een range vraag kan er een getal tussen 0 en 100 worden aangegeven door middel van een slider. Bij de antwoorden wordt er dan een range aangegeven

voor wat dit antwoord van toepassing is. bijvoorbeeld "0-50".

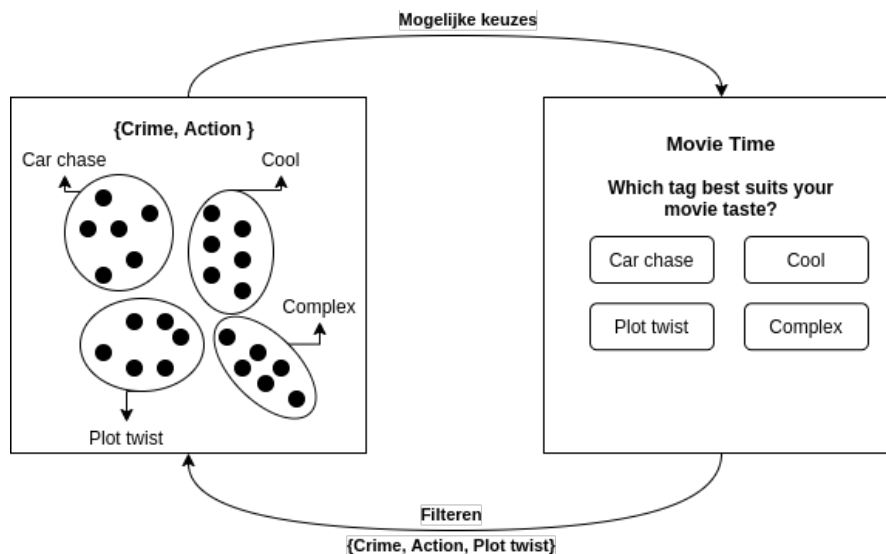
- Image: Net zoals bij de question type kan er 1 antwoord aangeduid worden, enkel worden de antwoorden nu voorgesteld door een afbeelding.
- Select: Hierbij kunnen meerdere antwoorden geselecteerd worden. Bij dit vraagtype moet er een "amount" worden meegegeven dat het aantal te selecteren antwoorden voorstelt.

Ieder mogelijk antwoord bevat een vector met negentien elementen die een of nul zijn. Iedere van de negentien elementen stelt een genre voor. Indien een antwoord een 1 voor een bepaalde genre bevat wilt dit zeggen dat dit antwoord aansluit bij dit genre. Tijdens het spel worden al de genre-vectors van de gekozen antwoorden elementsgewijs opgeteld. Aan het einde van het spel wordt de genre met de hoogste score gekozen als aanbevolen genre en gaat het spel verder naar de tag-dilemma's.

Listing 3 Voorbeeld van het JSON-vragen bestand

```
{
  "question": "Je ogen moeten nog even wennen aan het licht maar al snel
              wordt duidelijk waar je beland bent. Je bent in...",
  "type": "question",
  "answers":
  [
    {
      "id": 0,
      "genre": [1,1,1,0,0,0,0,1,1,0,0,0,1,0,1,0,1,1,0],
      "value": "een ruimeschip",
      "next": 2
    },
    {
      "id": 1,
      "genre": [1,1,1,0,0,0,0,1,1,1,0,1,0,1,1,0,0,1,0,0],
      "value": "een kasteel",
      "next": 2
    },
    {
      "id": 2,
      "genre": [1,1,0,0,1,0,1,0,0,0,0,0,0,1,0,1,0,1,1,1],
      "value": "een gevangenis",
      "next": 2
    },
    {
      "id": 3,
      "genre": [1,0,0,1,1,0,1,0,0,0,0,0,0,1,1,0,0,1,0,1],
      "value": "een cafe",
      "next": 2
    }
  ]
}
```

5.2 Tag-dilemma's

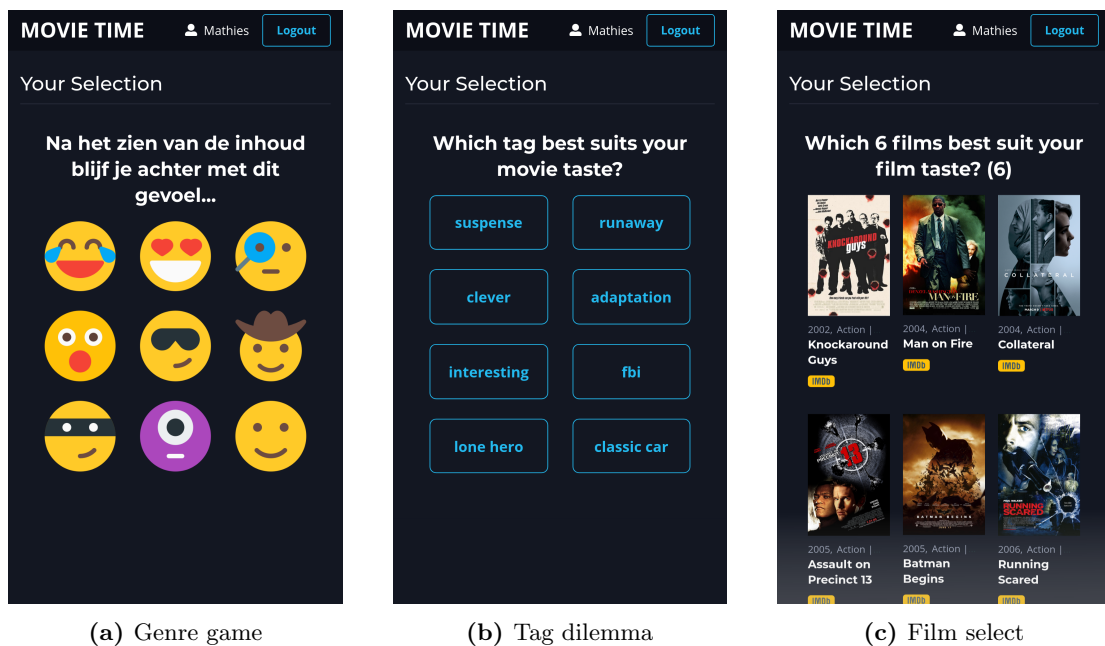


Figuur 5.1: Schematische werking van de tag-dilemma's

De tag dilemma's vormen de belangrijkste stap in het opbouwen van het filmprofiel. De info die hier verzameld wordt, vormt een voorstelling van de filmsmaak van een gebruiker. Het is belangrijk dat de gebruiker de controle over het systeem blijft behouden en zo goed mogelijk zijn filmsmaak kan benaderen. De tag-dilemma's moeten de gebruiker helpen om uit 1129 tags de tags te kiezen die de filmsmaak van de gebruiker het dichtst benaderen. Dit gebeurt door de gebruiker te laten navigeren door een overzicht van tags waarbij iedere keuze de volgende mogelijke tags beïnvloedt. In figuur 5.2b is het scherm te zien zoals dit in de applicatie wordt weergegeven. In figuur 5.1 is een schematische voorstelling van hoe dit juist in zijn werk gaat. De space met films wordt via een clustering algoritme opgedeeld in verschillende clusters. Van iedere cluster wordt de overheersende tags gezocht. Deze tags worden voorgesteld aan de gebruiker. Op basis van zijn keuze wordt de space met films gefilterd en begint de clustering opnieuw. Dit wordt herhaald tot er voldoende tags voor een accuraat filmprofiel gevonden zijn.

5.2.1 Implementatie

Na iedere keuze van de gebruiker worden de volgende stappen doorlopen. Als eerste wordt de geselecteerde tag aan de lijst met *filter_tags* toegevoegd. Vervolgens worden de films gefilterd op basis van de tot dan toe geselecteerde tags. Dit wordt gedaan door de *findNewSet* (listing 4) functie op te roepen. Deze functie krijgt als attributen de tot dan toe geselecteerde tags, het aantal films en de gefilterde lijst van al de mogelijke films. Het resultaat van deze functie is een lijst van films die tot dan toe aan de geselecteerde tags voldoen. Om films te vinden op basis van een lijst van tags maakt de *findNewSet* functie gebruik van de *tagSimilarity* functie die besproken werd



Figuur 5.2: Screenshots van het opbouwen van een filmprofiel

in sectie 4.2.

Listing 4 Algoritme voor het vinden van een nieuwe set van films

```
def findNewSet(filter_tags, amount, genre_movie_list):
    similar_movies = recommender.tagSimilarity(filter_tags, amount, genre_movie_list)
    all_tag_list = recommender.getTagList()
    vectors = pd.DataFrame(recommender.getTagVector())
    clusters = all_tag_list[all_tag_list['movieId'].isin(similar_movies)]
    index = clusters.index.to_list()
    vectors = vectors.iloc[index].to_numpy()
    return clusters, vectors
```

De volgende stap is het vinden van clusters in de set van mogelijke films. Iedere film wordt voorgesteld door een vector. Deze vectors worden gebruikt als input voor het clustering algoritme. De gevonden clusters stellen groepen van films voor die nauw samenhangende tags hebben. Voor de clustering wordt er gebruik gemaakt van K-Means clustering [21]. Na het vinden van de clusters wordt er voor iedere cluster de belangrijkste tag gezocht. Deze tags worden dan aan de gebruiker getoond zodat het proces weer van voor af aan opnieuw kan beginnen. De belangrijkste tag in iedere cluster wordt gevonden door voor iedere Cluster de 30 meest voorkomende tags eruit te filteren. Vervolgens worden al de tags die in meer dan één cluster voorkomen verwijderd. Op deze manier worden de algemene tags eruit gefilterd. De overblijvende tags zijn de tags die

de cluster uniek maken. Als laatste stap wordt dan voor iedere cluster de tag gekozen die het meeste voorkomt. Deze tags worden vervolgens aan de gebruiker getoond. Het algoritme hiervoor is te zien in listing 5.

Listing 5 Algoritme voor het vinden van de belangrijkste tag in een cluster

```
def findClusterGround(data,base):
    tags = data[['Cluster', 'tag_list']].groupby('Cluster').agg('sum')
    tags['tag_count'] = tags['tag_list'].apply(
        lambda x: [e[0] for e in Counter(x).most_common(30)]
    )
    cluster_ground = []
    all_tags = []
    for index, row in tags.iterrows():
        all_tags += row['tag_count']
    tag_intersection = [k for k, v in Counter(all_tags).items() if v > 1]
    for index, row in tags.iterrows():
        cluster_ground.append(
            np.setdiff1d(np.setdiff1d(row['tag_count'], tag_intersection), base)
        )
    return cluster_ground
```

5.3 Film selectie

De laatste stap tot een volledig filmprofiel is het selecteren van enkele films die aansluiten bij de filmsmaak. Op basis van de gekozen tags wordt er een lijst van films getoond. Hieruit kiest de gebruiker vervolgens zijn zes favoriete films. Dit scherm is te zien in figuur 5.2c. Na het kiezen van de films is het profiel voor deze gebruiker compleet. Het profiel bestaat nu uit een aantal tags en zes films die de filmsmaak van de gebruiker benaderen.

5.3.1 Implementatie

De lijst met geselecteerde tags van de gebruiker wordt als input geven aan het *tagSimilarity* algoritme, dat te zien is in listing 2. Dit algoritme krijgt ook een gefilterde lijst van films mee. Deze gefilterde lijst bevat enkel de films van het gekozen genre. Het resultaat zijn vijftien films die aan zowel het genre als de gekozen tags voldoen. Uit deze vijftien films moet de gebruiker vervolgens zes films kiezen. De gebruiker krijgt van iedere film de titel, het genre en een coverfoto te zien. Ook voorzien we een link naar de IMDB pagina van de film waar er meer gedetailleerde info of de film te zien is.

Omdat het ophalen van de coverfoto's op het internet een tijdrovende taak is, worden de coverfoto's opgeslagen in onze eigen database. Indien de cover nog niet aanwezig is in deze database wordt de coverfoto opgehaald van op IMDB en opgeslagen in de database.

Hoofdstuk 6

Groepsvoorspelling

Nu het persoonlijk filmprofiel van ieder lid compleet is kan er een groepsvoorspelling gemaakt worden. Voor het vinden de groepsvoorspellingen vertrekt het algoritme van een lijst met aanbevolen films voor iedere gebruiker. In sectie 6.1 wordt toegelicht hoe er voor iedere gebruiker individueel op zoek gegaan wordt naar een lijst van aanbevolen films. In sectie 6.2 wordt vervolgens besproken hoe deze individuele lijsten gebruikt worden om een groepsvoorspelling te doen.

6.1 Filtering

Als eerste wordt er gezocht naar een lijst van aanbevolen films voor iedere gebruiker. Om deze lijst te vinden wordt er gebruik gemaakt van het persoonlijk filmprofiel. Er zijn twee manieren waarop deze lijst gevonden kan worden. In de eerste methode wordt er gebruik gemaakt van de films op het profiel. Op basis van deze films wordt er met het algoritme uit listing 1 een lijst van vijftig gelijkende films gezocht.

Bij dit algoritmen wordt er gebruik gemaakt van de zes films die iedere gebruiker in de laatste stap van de profielopbouw geselecteerd heeft. Deze lijst van films wordt dan in de volgende stap verder gebruikt. Door gebruik te maken van deze tags en het algoritme uit listing 2 kan er voor iedere gebruiker een lijst met vijftig aanbevolen films gevonden worden.

6.2 Common-ground

In de vorige stap werd er voor iedere gebruiker een lijst van vijftig aanbevolen films gevonden. Als eerste stap wordt er gekeken of er een overlap tussen deze films bestaat. Indien dit het geval is zijn deze films aanbevolen voor al de leden in de groep en is de groepsvoorspelling gevonden. Dit gaat echter enkel het geval zijn wanneer de filmsmaak van de gebruikers erg gelijkend is.

Wanneer deze overlap er niet is wordt er gezocht naar de common-ground in de filmsmaak van de groep. De common-ground van een groep is een combinatie van tags die de filmsmaak van de groep weergeeft. Voor het vinden van de common-ground wordt er voor iedere gebruiker een lijst gemaakt met al de tags die meerdere keren in zijn lijst van voorspelde films voorkomen. Dit kan gezien worden als de uitgebreide filmsmaak van een gebruiker. Vervolgens wordt er een doorsnede genomen van deze uitgebreide filmsmaken. De tags in deze doorsnede stelt de gemeenschappelijke filmsmaak voor en kan net zoals bij de filmsmaak van een gebruiker, gebruikt worden om aanbevolen films te zoeken.

Een voorbeeld van een dergelijke common-ground is *action*, *sport*, *plot twist*, dit zou dus betekenen dat het gezelschap een gemeenschappelijke interesse heeft voor films die *action*, *sport* en een *plot twist* bevatten.

Wanneer deze common-ground gevonden is wordt er opnieuw gebruik gemaakt van het algoritme uit listing 2 om een lijst van films te vinden op basis van deze tags.

Deel III

Validatie en discussie

Hoofdstuk 7

Onderzoeksmethode

Zoals beschreven in sectie 3.4 is het prototype gebouwd als een Django applicatie. Dit maakt het mogelijk om het prototype eenvoudig te deployen en dit aan gebruikers te laten testen. Er werd gebruik gemaakt van een Digital Ocean droplet om de implementatie te hosten op het domein movieTime.be.

Een zesdelige vragenlijst werd opgesteld om feedback over het platform te verzamelen. Tijdens de vragenlijst worden de gebruikers ook wegwijs gemaakt op het platform en wordt hun stap voor stap uitgelegd welke acties ze moeten uitvoeren op het platform. Het eerste deel van de vragen gaat over het individueel kijkgedrag van de gebruiker. Hier wordt gepeild naar hoe de gebruiker gewoonlijk op zoek gaat naar een film en hoe gemakkelijk ze een gepaste film weten te vinden. In het volgende deel wordt er dieper in gegaan op het kijkgedrag van de gebruiker in groep. Hier proberen we te achterhalen hoe de gebruikers gewoonlijk op zoek gaan naar een film in groep en waar de pijnpunten zitten. Vervolgens wordt er aan de gebruiker gevraagd om een account aan te maken op Movie Time en hun persoonlijk profiel te vervolledigen. Er wordt dan aan de gebruiker gevraagd om het doorlopen proces te beoordelen en een score te geven op het resultaat. Het resultaat zijn de tags en de films die hun filmsmaak voorstellen. In de volgende stap gaan ze de groepsvoorspelling beoordelen hiervoor genereren ze eerst een voorspelling op basis van films en vervolgens op basis van tags. De vragenlijst vraagt eerst om de resultaten individueel te beoordelen en vervolgens de voorspelde films in groep te bespreken en als groep te beoordelen. Dit wordt op dezelfde manier gedaan voor beide algoritmen zodat deze onderling vergeleken kunnen worden. De laatste stap van de vragenlijst verzamelt nog de algemene mening van de gebruikers tegenover een platform zoals Movie Time. De volledige vragenlijst is te zien in bijlage A.

7.1 Deelnemers

Via het vak Human-AI interaction aan de U Hasselt en ander kanalen zoals sociale media werd er een oproep gedaan om deelnemers voor het onderzoek te vinden. Deelnemen kon enkel in groep, overigens is het belangrijk dat de groepen onderling kunnen communiceren om de groepsvoorspelling correct te kunnen beoordelen. Zestien personen

hebben het onderzoek voltooid, dit waren zeven groepen van twee of drie personen. De deelnemers bestaan uit tien studenten informatica en zes personen zonder informatica achtergrond. Het voltooien van de vragenlijst nam ongeveer 20 minuten in beslag.

Hoofdstuk 8

Analyse en Resultaten

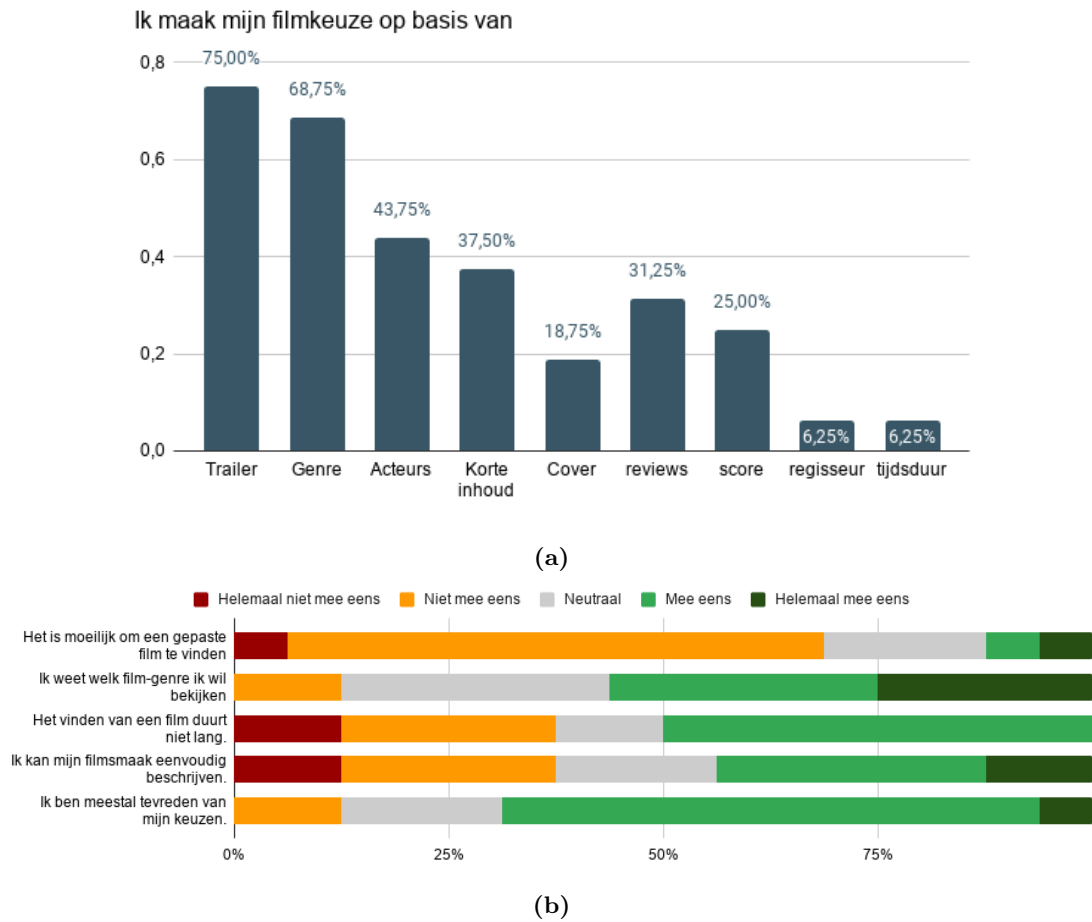
In deze eerste formatieve evaluatie worden er twee belangrijke dingen afgetoetst. Ten eerste wordt met dit onderzoek de methode om via tag-dilemma's een filmsmaak vast te leggen geëvalueerd. Als tweede willen we bekijken hoe goed de groepsvoorspellingen werken en welk van de twee algoritmen het beste scoort. Om te weten te komen wat het kijkgedrag van de deelnemers is zonder het gebruik van het systeem kijken we in het eerste deel naar het algemene kijkgedrag individueel en in groep.

8.1 Kijkgedrag individueel

De resultaten van het individuele kijkgedrag zijn als volgt. In figuur 8.1b is te zien dat de deelnemers over het algemeen weinig problemen hebben om alleen een film te kiezen. Ze weten welk genre ze willen zien en het duurt niet lang voor ze een film gevonden hebben. Na het kijken van de film zegt de meerderheid ook effectief tevreden te zijn met hun keuze. De elementen die de doorslag geven bij het kiezen van een film worden getoond in figuur 8.1a. Hier is te zien dat de meeste deelnemers hun keuze vooral laten afhangen van de trailer en het genre. Veel minder belangrijk blijkt de tijdsduur en de regisseur. Er werd ook aan de deelnemers gevraagd om hun filmsmaak te omschrijven. In al de gevallen werd de filmsmaak omschreven door een of meerder genres die in enkele gevallen werden aangevuld door inhoudelijke elementen of favoriete films.

8.2 Kijkgedrag in groep

Nog voor de deelnemers gebruik maken van het Movie Time platform wordt er gevraagd naar hun film-kijk ervaring in groep. 43.8% van de deelnemers geeft aan maandelijks een film in groep te bekijken, voor 37.5% is dit zelfs wekelijks het geval. Deze groepen bestaan in de meeste gevallen (62.3%) uit twee personen, 18.8% uit drie personen en 12.5% uit vier personen. Wanneer er naar de resultaten in figuur 8.2 wordt gekeken is er te zien dat de meerderheid moeilijkheden heeft om een film in groep te vinden. Ook wordt er aangegeven dat het lang kan duren voor er effectief een film gevonden wordt

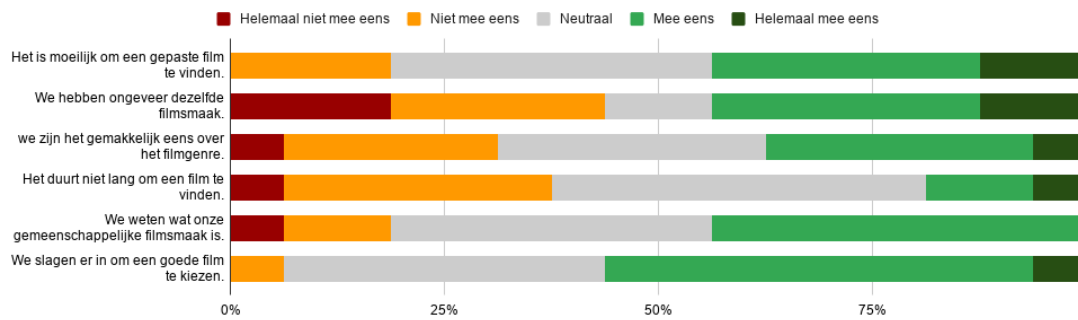


Figuur 8.1: Resultaten individueel kijkgedrag

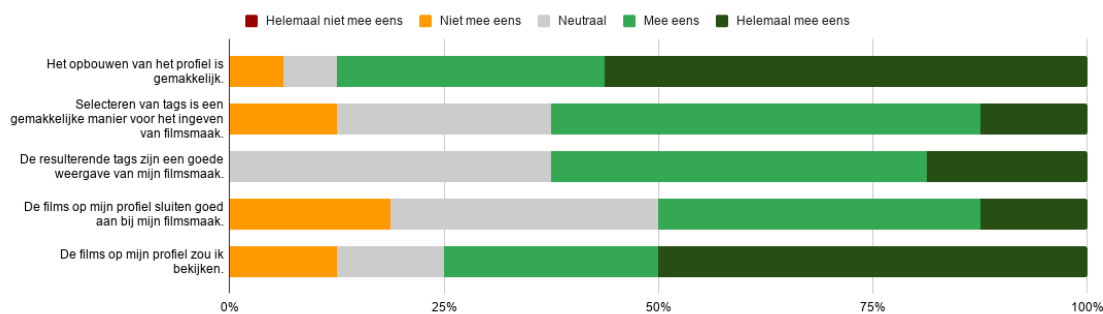
8.3 Persoonlijk profiel

In deze stap vervolledigen de deelnemers hun filmprofiel. Nadat hun filmprofiel compleet is krijgt iedere deelnemer een lijst van aanbevolen films te zien. Er werd aan de deelnemers gevraagd om zowel het proces tijdens de opbouw van het profiel te beoordelen als de resultaten.

In figuur 8.3 is te zien dat het opbouwen van het profiel als gemakkelijk wordt ervaren. De deelnemers geven aan dat het selecteren van tags een gemakkelijke manier is voor het ingeven van een filmsmaak en dat zowel de resulterende tags als voorspelde films een goede benadering zijn voor hun filmsmaak.



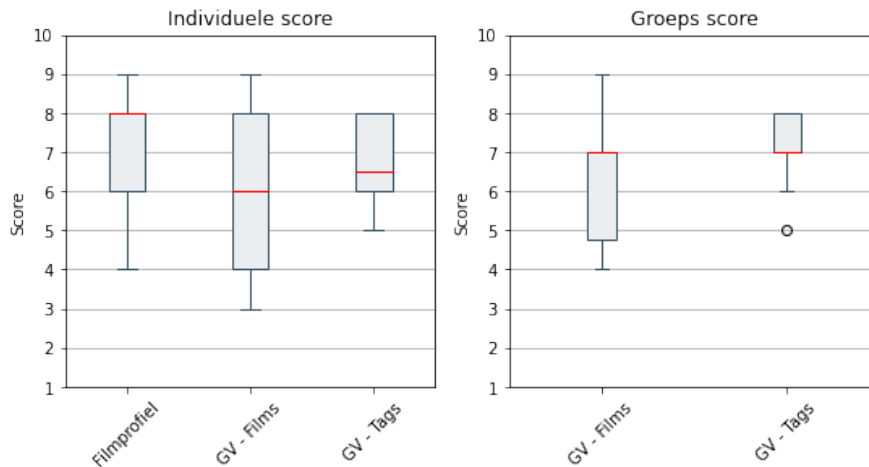
Figuur 8.2: Resultaten kijkgedrag in groep



Figuur 8.3: Resultaten opbouwen filmprofiel

8.4 Groepsvoorspelling

De groepsvoorspelling bestaat uit twee delen, het algoritme dat werkt op basis van films en het algoritme dat werkt op basis van tags. De gebruikers worden gevraagd om beide algoritmes te testen. Voor ieder algoritme werden dezelfde vragen gesteld. Eerst wordt er aan iedere gebruiker gevraagd om de resultaten individueel te beoordelen en vervolgens mogen de deelnemers de voorspelde films als groep evalueren en een score toewijzen. In figuur 8.4 is links de score te zien die de gebruikers individueel gaven en rechts de groepscore. De mediaan van de scores wordt vergeleken. Het algoritme op basis van films krijgt een individuele score van zes en een groepscore van zeven. Het algoritme op basis van tags krijgt een individuele score van 6.5 en ook een groepscore van 7. Op basis van de box-plots is te zien dat de resultaten van het algoritme op basis van tags veel minder verspreid liggen dan bij die op basis van films. Bij beide algoritmen gaf 81.3% van de deelnemers aan dat er in de lijst een film zat die ze als groep zouden bekijken. 18.8% gaf aan dat er misschien een film tussen zat die ze zouden kijken.



Figuur 8.4: Score filmprofiel en groepsvoorspelling

8.5 Algemeen

In de laatste stap van de vragenlijst werd er gevraagd naar de algemene ervaring met het platform. Hierbij gaf 93.8% aan dat het vinden van een gepaste film in groep sneller zou gaan dankzij de voorspelde films van het systeem. 56.3% gaf aan in de toekomst een app zoals Movie Time te willen gebruiken om in groep een film te vinden. 37.5% gaf aan in de toekomst misschien een app zoals Movie Time te willen gebruiken. De laatste vraag van de vragenlijst is een open vraag waar de deelnemers hun overige feedback kunnen achterlaten. Hieronder volgt een lijst van de belangrijkste feedback die verzameld werd:

- Meer variëteit in de tags bij de tag-dillema's
- Het zou ook handig zijn als je op je profiel een aantal van je favoriete films kan ingeven.
- Meer info tonen bij de films, zoals beschrijving.
- Common-ground tonen is goed zodat de gebruiker ziet waar overlap ligt
- Meer aangeven waarom bepaalde films bij de groepssuggestie aanwezig is.
- Misschien helpt het aangeven van een humeur/ gevoel? En een geslacht en leeftijd?
- Misschien was het handig geweest om ook te zeggen dat je een bepaalde categorie echt niet wilt.

Hoofdstuk 9

Discussie

Binnen deze discussie wordt er gekeken of de resultaten voldoende overtuigend zijn om dit systeem verder uit te bereiden met de verzamelde feedback. Het is belangrijk op te merken dat het onderzoek op een beperkte groep (zestien personen verdeeld over zeven groepen) werd uitgevoerd. Dit zijn dus geen significante resultaten maar slechts eerste bevindingen.

9.1 Filmprofiel

De resultaten in sectie 8.1 geven aan dat meer dan de helft van de deelnemers weet welk filmgenre ze willen kijken. Hieruit kan afgeleid worden dat de genre-game die toegelicht wordt in sectie 5.1 in een grote meerderheid van de gebruikers niet van toepassing is. De genre-game is geen essentieel onderdeel meer van het systeem. Meer essentieel is het verfijnen van de filmsmaak. Deze verdere verfijning gebeurt op basis van de tag-dilemma's. Uit de resultaten van sectie 5 is duidelijk te zien dat de deelnemers het opbouwen van hun profiel op basis van tags als gemakkelijk ervaren. Alle gebruikers zijn het eens dat de resulterende tags op hun profiel een goede representatie van hun filmsmaak is. Ook de films die we voorspellen op basis van deze tags worden door de meeste personen aanzien als goede voorspellingen. Op basis van deze resultaten wordt geconcludeerd dat inhoudelijke film-tags kunnen dienen als een goede abstractie van de filmsmaak van een individu.

Een tag kiezen uit een lijst met slechts acht voorgestelde tags blijkt voor veel deelnemers te beperkend. Hier zou de mogelijkheid moeten zijn om meerdere tags te selecteren of de tags te vernieuwen wanneer er geen tags aansluiten bij de filmsmaak.

Bij het filteren van tags om voor te stellen bij het tag-dilemma wordt er een clustering gedaan van al de films in een bepaald genre zonder dat deze films op een andere manier verder gefilterd of beperkt worden. Bij genres zoals Comedy' waar er een grote variatie van types film binnen dat bepaald genre is, geeft het selecteren van tags in sommige gevallen niet het gewenste resultaat. Dit zou opgelost kunnen worden door de films eerst verder te filteren alvorens met de tag dilemma's te beginnen. Deze verdere filtering zou kunnen zijn op basis van sub-genre, score... Hierbij moet er ook rekening gehouden wor-

den dat een verdere filtering ook invloed kan hebben op de controle en verklaarbaarheid van het systeem. Het is hier dus aangewezen om verder onderzoek te doen naar een gulden middenweg tussen accuraatheid en verklaarbaarheid.

Zoals bij de implementatie besproken werd, worden de voorgestelde tags gevonden op basis van een clustering algoritme. Het aantal clusters en dus ook voorgestelde tags ligt vast. Binnen dit onderzoek werden acht clusters gekozen. De meest overheersende tag van iedere cluster wordt dan aan de gebruiker voorgesteld. Door deze indeling in een vast aantal clusters, zullen groepen van films die minder vertegenwoordigd zijn overheerst worden door groepen van films die in grotere hoeveelheden aanwezig zijn. Als we terug komen op het voorbeeld van de komedie films wordt het volgende probleem opgemerkt. Het genre komedie kan opgedeeld worden in meer dan tien subgenres (Action comedy, Romantic comedy,...). Door een opdeling te maken in een vast aantal clusters gaan meerdere subgenres in dezelfde cluster terecht komen en gaat de overheersende tag in zo een cluster altijd een tag van het subgenre zijn met de meeste films. Hierdoor is het mogelijk dat de gebruiker bij bepaalde genres door het systeem in een richting geduwd wordt die niet de meest optimale keuze voor de gebruiker voorstelt. Dit kan opgelost worden door niet met een vast aantal clusters te werken maar het optimale aantal clusters te bepalen op basis van de intra cluster variantie. Hier moet wel een afweging gemaakt worden tussen het vinden van een optimale oplossing en de performance van het systeem. Aangezien de clustering en het vinden van de tags in realtime gebeurt moet er rekening gehouden worden dat dit geen lange wachttijd voor de gebruiker met zich meebrengt.

9.2 Groepsvoorspelling

De focus van deze thesis ligt op het vinden van een groepsvoorspelling. Voor het vinden van de groepsvoorspelling werden er twee verschillende aanpakken geïmplementeerd die met elkaar vergeleken worden. Eerst worden de verschillende onderdelen van de groepsvoorspelling afzonderlijk besproken. Vervolgens worden beide algoritmen vergeleken met elkaar.

9.2.1 Voorspelling op basis van films

Zoals al eerder toegelicht gebruikt dit algoritme de films die de deelnemers op hun filmprofiel geselecteerd hebben. Wanneer er gekeken wordt naar de persoonlijke score die de deelnemers aan deze voorspelling geven is het duidelijk dat deze scores erg variërend zijn. De scores liggen tussen de drie en de negen op tien. De mediaan bij dit algoritme is een zes. Vervolgens wordt er naar de groepscore gekeken waar ook een grote variatie te zien is. Deze score heeft een mediaan van zeven. Net zoals de verwachtingen is de score van de groep hoger dan de individuele score. Een goede groepsvoorspelling sluit bij iedereen van de groep een beetje aan. Het is dus logisch dat de groepsvoorspelling

een lagere score krijgt dan de persoonlijke voorspelling. Het is goed dat de score stijgt wanneer de deelnemers als groep de films evalueren. Dit betekent dat de voorspellingen beter worden wanneer iedereen aan elkaar heeft toegelicht wat hun respectievelijke filmsmaken zijn. Ondanks de grote variëteit in de score geeft toch 81.3% aan dat ze als groep een film uit de lijst zouden bekijken.

Bij de feedback over het filmprofiel werd aangegeven dat bij het selecteren van de films voor op het filmprofiel er niet altijd voldoende opties beschikbaar zijn. Dit zorgt ervoor dat er films gekozen moesten worden die niet volledig bij de filmsmaak aansloten. Op dit moment worden er vijftien films voorgesteld waar de gebruiker er zes van moet kiezen. Indien er geen zes films zijn die aansluiten bij de smaak van de gebruiker moet er de mogelijkheid zijn om met minder films verder te gaan of nieuwe voorstellen te krijgen waaruit de films gekozen kunnen worden. Tijdens het onderzoek hebben sommige gebruikers minder optimale films moeten kiezen. Dit beïnvloedt de prestaties van het algoritme en kan mee aan de oorzaak liggen van de grote variëteit in de scores die aan dit algoritme werden toegewezen.

9.2.2 Voorspelling op basis van tags

Bij de voorspelling op basis van tags worden de tags van op het filmprofiel gebruikt om een groepsvoorspelling te doen. De individuele score van het tag-algoritme ligt tussen de vijf en de acht met als mediaan 6,5. Net zoals bij het vorige algoritme ligt de groep score hoger dan de individuele score. De groepscore ligt tussen de zes en de acht met een mediaan van zeven. Ook bij dit algoritme geeft 81.3% aan dat er een film aanwezig is die ze als groep zouden kijken. Bij dit algoritme hebben de gekozen tags een grote invloed op de accuraatheid van het resultaat. Bij de resultaten van de tags op het filmprofiel (figuur 8.3) is te zien dat hier niemand aangaf het oneens te zijn met de tags die hun filmsmaak samenvatten. De tags lijken dus een goede basis om op verder te werken bij het bepalen van een groepsvoorspelling. Het feit dat niemand het oneens was met de tags kan ook verklaren waarom er een minder grote variatie in de scores van dit algoritme is.

9.2.3 Common-ground

Zoals reeds toegelicht gaat in beide gevallen het algoritme eerst op zoek naar een doorsneden in de films en indien deze leeg is, wordt er gezocht naar een common grond. Deze common-ground wordt dan gebruikt om voorspellingen te doen voor de groep. Op basis van de resultaten zijn er twee problemen met deze common-ground opgevallen die kunnen leiden tot slechte groepsvoorspellingen. In sommige gevallen bestaat de common-ground uit negatieve tags. Een voorbeeld hiervan zou zijn: ['Bad acting', 'to long', 'boring']. Wanneer er een common-ground gevonden wordt met negatieve tags kan het niet anders dat het resultaat ook negatief gaat zijn. De oorzaak van dit probleem is dat er op dit moment geen enkele filtering gebeurt op de tags en aangezien tags door de community zijn toegewezen zijn er zowel positieve als negatieve tags. Een mogelijke oplossing zou zijn om al de negatieve tags weg te laten maar dit kan een negatief effect hebben op de

accuraatheid van het systeem. Wanneer al de negatieve tags er uit gefilterd worden zou een film met de tags: ['action', 'car chase', 'lonely hero'] hetzelfde score als een film met de tags ['bad acting', 'bad plot', 'boring', 'action', 'car chase', 'lonely hero']. Er gaat dus belangrijke info verloren. Ook is het concept negatieve tags moeilijk eenduidig te bepalen. Tags die voor de ene gebruiker als negatief ervaren wordt, kan voor de andere gebruiker net onderdeel van hun filmsmaak zijn.

Een ander mogelijk probleem is een te beperkte common-ground. Wanneer de gevonden common-ground slechts enkele tags bevat gaat het uiteindelijke resultaat zo uiteenlopend zijn dat er geen geslaagde groepsvoorspelling is. In het systeem zien we een groep waarbij de common-ground enkel bestaat uit de tag ['action'] deze groep krijgt dus gewoon de vijftien eerste actie films te zien die het systeem vindt.

Op dit moment is de common-ground de enigste filter parameter. Dit werd gedaan om het resultaat van filtering op basis van common-ground goed te kunnen evalueren. Wanneer dit systeem in de toekomst gebruikt wordt, is het aan te raden om de common-ground uit te bereiden op basis van andere parameters. Een andere optie is om de gebruikers hun filmprofiel verder te laten verfijnen wanneer de common-ground te beperkt is. Hoe uitgebreider het persoonlijk profiel van iedere gebruiker hoe meer kans dat er een voldoende uitgebreide common-ground gevonden wordt.

9.2.4 Vergelijking

Op basis van de vergelijking van de scores van beide algoritmen, lijkt het verschil minimaal. De mediaan van de individuele score ligt iets hoger bij het tags algoritme (verschil van 0.5) en is hetzelfde bij de groep score. Wel is er een groot verschil in spreiding van de score te zien. Bij het algoritme op basis van films zijn de resultaten heel variabel. In sommige gevallen scoorde het algoritme heel goed in andere gevallen scoorde het algoritme ondermaats. Bij het algoritme op basis van tags is deze spreiding veel minder uitgesproken. De score van het tag-algoritme heeft een minder grote spreiding dan dat van het film algoritme. Deze minder grote spreiding maakt het algoritme voorspelbaarder en op basis van de resultaten die nu beschikbaar zijn de beste keuze voor het systeem. Bij het film-algoritme werd al aangegeven dat de beperkte keuze van films een grote invloed heeft gehad op de prestaties van dit algoritme. Dit is een belangrijk inzicht dat opgelost kan worden in een volgende versie. De aangehaalde problemen en de beperkte testen maken het in dit stadium dan ook te vroeg om één van de twee algoritmen te verkiezen boven het andere.

9.3 Algemene discussie

Het is belangrijk nog even stil te staan bij het feit dat de test gebruikers de score gegeven hebben op basis van de coverfoto en de titel van een film. Indien de gebruikers meer info over een film wilden, konden ze de IMDB-knop gebruiken om de IMDB-pagina van de film te bekijken. Er werd niet aan de gebruikers gevraagd om de film(s) ook effectief te bekijken. In sectie 2.4 wordt aangegeven hoe de accurateheid van een aanbe-

velingssysteem beoordeeld kan worden. Hierbij wordt ook aangegeven dat serendipiteit een belangrijke parameter is in de evaluatie van aanbevelingssysteem. Serendipiteit is een maatstaf voor hoe verrassend een succesvolle aanbeveling is. Het is mogelijk dat er films voorspeld werden die voor de gebruiker op het eerste opzicht lijken als een slechte voorspelling, maar wanneer men de films zou bekijken een aangename verrassing zouden zijn.

Dit onderzoek leert ons nog enkele andere elementen die effect hebben op de evaluatie van het systeem. Uit de resultaten blijkt dat de keuze voor een film vooral op basis van de trailer en het genre gemaakt wordt. In het systeem krijgen de gebruikers de titel, cover en genres van een film te zien. De trailer is niet opgenomen in het systeem maar zoals de gebruikers aangeven zou het opnemen van de trailer het keuze en evaluatie proces kunnen vergemakkelijken.

Hoofdstuk 10

Toekomstig werk

In deze thesis werd een eerste proof of concept van het Movie Time platform ontwikkeld. De testen met deze proof of concept werden positief onthaald maar het is in dit stadium nog te vroeg om te kunnen besluiten welk van de twee geïmplementeerde algoritmen het beste werkt en hoe accuraat deze algoritmen zijn. Er werd een eerste aanzet gegeven en belangrijke feedback verzameld voor het verder ontwikkelen van het platform. De verzamelde feedback werd al kort aangehaald bij de resultaten en de discussie. Hieronder worden de belangrijkste elementen voor de volgende versie van het platform verder besproken.

Accuraatheid Er zijn verschillende elementen die invloed hebben op de accuraatheid van de voorspellingen. Bij de discussie werden er al enkele kleine wijzigingen besproken die een grote invloed kunnen hebben op de prestatie van het algoritme. Dit gaat meer bepaald om het verfijnen van het tag-dilemma proces zodat de voorgestelde tags een betere representatie weergeven van de filmsmaak én het uitbreiden van de genome tag filtering met andere state-of the art algoritmen om een hybride aanpak te vormen die mogelijk de nadelen van de genome tag filtering kan verminderen. Ook zijn er verschillende hyper-parameters in het systeem die aangepast kunnen worden om een grotere accuraatheid te bekomen.

Transparantie Bij onze implementatie staat de gebruiker centraal. Het is de bedoeling dat de gebruiker al de controle over het systeem heeft en ten alle tijden zijn gebruikersprofiel kan aanpassen of updaten. Wat er in deze proof of concept nog mist is transparantie in de voorspelling van films. Hoe meer inzicht de gebruiker in de keuzes van het systeem heeft hoe beter hij de voorspellingen gaat begrijpen. Op dit moment heeft de gebruiker het inzicht in de data die aan het systeem wordt gevoed, namelijk de data van op zijn persoonlijk filmprofiel. Echter nadat er een groepsvoorspelling gedaan wordt, ontbreekt de informatie waarom bepaalde films een goede match zouden zijn voor de groep. Dit inzicht zou kunnen zorgen voor een verhoogde tevredenheid. Toekomstig werk zou deze inzichten kunnen toevoegen aan het bestaande systeem.

Evaluatie Binnen dit onderzoek is er een formatieve evaluatie gebeurd op een beperkte groep van deelnemers. Verdere evaluatie zal moeten aantonen hoe accuraat het systeem is. De evaluatie gebeurde nu vooral op groepen van twee personen (op een uitzondering van vier personen na). In een toekomstige evaluatie zal er verder gefocust moeten worden op het effect van een grotere groep op de accuraatheid van de voorspelling.

Uitbreidingen Het systeem omvat nu een vlot werkend proof of concept waarbij op een interactieve manier een gebruikersprofiel kan opgebouwd worden door middel van tag dilemma's. Deze aanpak kan in de toekomst verder toegepast worden op andere data sets zoals boeken, reizen of zelfs reclame voorkeuren. Ook zonder de groepsvoorspelling is dit een interessante manier om een gebruikersprofiel op een transparante gebruiksvriendelijke manier op te bouwen. Dikwijls worden gebruikersprofielen en aanbevelingen op een black-box manier gedaan. Informatie wordt op de achtergrond verzameld door gebruikers hun acties te volgen en vervolgens worden er op basis van deze informatie een voorspelling aan de gebruiker gedaan. De gebruiker kan deze black-box voorspellingen niet gemakkelijk zien of aanpassen. Een implementatie van een gebruikersprofiel op basis van tag-dilemma's en genome tags zou in de toekomst een basis kunnen zijn voor aanbevelingen waarbij de gebruiker en zijn noden centraal staan.

Hoofdstuk 11

Conclusies

Binnen deze masterproef is er onderzoek gedaan naar een *user centered groepsaanbevelingssysteem voor films*. Hiervoor werd een proof of concept opgesteld van een groepsaanbevelingssysteem waar de gebruiker centraal staat. In de praktijk hebben veel aanbevelingsalgoritmen niet als doel de gebruiker te helpen of het leven gemakkelijker te maken. Een groot aantal aanbevelingsalgoritmen hebben een verborgen agenda. Namelijk zorgen dat je een product koopt, dat je zo lang mogelijk klant blijft... Bij deze algoritmen wordt er bewust geen controle aan de gebruiker gegeven. De gebruiker weet niet op welke data deze algoritmes werken of waarom bepaalde voorspellingen gedaan worden. Dit maakt het voor de gebruiker moeilijk om voorkeuren bij te stellen of zijn smaak volledig aan te passen. In de praktijk gebeurt het vaak dat een gebruiker zijn smaak wilt aanpassen aan het gezelschap of het moment. Daarom de keuze om in deze thesis te focussen op user-centered-recommendation. Ook wordt er in de praktijk relatief weinig aandacht gegeven aan groepsaanbevelingen. Dus *het doel van deze thesis was een proof of concept te ontwikkelen van een groepsaanbevelingssysteem waar de gebruiker centraal staat*. Als usecase werd er gekozen voor het vinden van een film in groep maar de besproken principes en algoritmen kunnen eveneens in andere usecases gebruikt worden.

De meeste state-of-the-art aanbevelingsalgoritmen vertrekken vanuit collaborative filtering waarbij er op basis van het gebruikersprofiel, dat meestal de kijkgeschiedenis en ratings van een gebruiker bevat, andere gelijkende gebruikers gezocht worden. Op basis van het kijkpatroon van deze gebruiker wordt er dan een voorspelling gedaan. Een andere methode is content-based filtering waarbij er vanuit de inhoud van een film andere films aanbevolen worden. Het Movie Time systeem is een implementatie van een content-based filtering systeem waarbij het in tegenstelling tot een coloboratieve filtering algoritme niet perse nodig om de gebruiker voor een langer tijd te volgen. Het is echter wel nodig om een gebruikersprofiel te hebben met een representatie van de filmsmaak van de gebruiker. Indien we de gebruiker controle willen geven bij het aanmaken van dit gebruikersprofiel, is het belangrijk op zoek te gaan naar *een eenvoudige manier om de filmsmaak in te geven in het systeem*. Simpelweg aan gebruikers vragen om hun film-

smaak te omschrijven zou de eenvoudigste aanpak zijn, maar deze aanpak heeft enkele tekortkomingen. Gebruikers kunnen vaak niet allemaal eenvoudig omschrijven wat hun filmsmaak is. Dit blijkt ook uit de resultaten van de vragenlijst waar er aan de gebruikers gevraagd werd om hun filmsmaak te omschrijven. Wanneer gebruikers hun filmsmaak omschrijven is dit meestal slechts het genre of een favoriete regisseur of acteur. Dit is te beperkt om als basis voor een aanbevelingsalgoritmen te gebruiken. Ten tweede is deze beschrijving van filmsmaak niet gestructureerd genoeg om te kunnen vergelijken met films. Op basis van een vrij tekstveld is dit bijna onmogelijk. Om deze problemen te omzeilen werd er gebruik gemaakt van tag genomes. De Movielens dataset bevat voor iedere film een lijst van tags. Deze tags bevatten niet enkel inhoudelijke kenmerken maar ook meer abstracte kenmerken zoals prijzen, scores,... Deze gestructureerde data geeft de mogelijkheid om films onderling op basis van hun inhoud te vergelijken. Verschillende onderzoeken [14, 8] toonde aan dat het mogelijk is om een content-based aanbevelingsalgoritme te maken op basis van deze tag genome.

Net zoals een film gedefinieerd wordt door een lijst van tags, willen we de filmsmaak van een gebruiker voorstellen door een lijst van tags. De gebruiker tags laten kiezen uit 1129 mogelijke tags is echter een onbegonnen opdracht. Daarom introduceren we *de tag-dilemma's*. *Dit is een interactief spelletje waarbij de gebruiker op een snelle manier door de tags navigeert en zo zijn smaakprofiel opbouwt*. Tijdens het navigeren worden op basis van iedere keuze nieuwe tags gegenereerd. Deze kunnen de gemaakte keuzes verder verfijnen. Dit zorgt er voor dat men met een tiental klikken een erg specifiek smaakprofiel kan opbouwen. Nadat we een smaakprofiel van de gebruikers hebben kunnen we via de gekende manieren een aanbeveling zoeken voor deze gebruikers.

Tot nu toe hebben we gefocust op een aanbevelingsalgoritme waar de gebruiker centraal staat. In de volgende stap wordt het uitgebreid naar het zoeken van groepsaanbevelingen. Zoals beschreven in sectie 2.2 is er de laatste jaren al heel wat onderzoek gebeurd naar Het zoeken van groep aanbevelingsalgoritmen. Daarbij wordt er vaak vertrokken van gebruikersprofielen die opgebouwd zijn met data waar de gebruiker geen controle over heeft. Er wordt geen rekening gehouden met het feit dat je filmsmaak kan variëren afhankelijk van de groep waarin je de film kijkt. Ons groepsaanbevelingsalgoritme vertrekt dan ook vanuit het flexibele gebruikersprofiel dat de gebruiker afzonderlijk kan aanpassen voor iedere groep. Hoewel de data waarop we werken fundamenteel verschillend is van de eerdere methoden, volgen we toch ongeveer dezelfde aanpak als de eerder besproken aanpakken. Er werd eerst naar *een overlap in de persoonlijke voorspellingen van iedere gebruiker gezocht*. *Indien er geen sprake was van een overlap aan films, werd de common-ground gezocht*. Dit is een lijst van tag genomen die aan al de gebruikers hun smaak voldoet. *Deze common-ground vormt een pseudo-gebruiker die de gemeenschappelijke smaak van al de gebruikers vertegenwoordigt*. Voor deze pseudo-gebruiker kan er dan net zoals bij gewone gebruikers een voorspelling gedaan worden.

Dit onderzoek is erin geslaagd om een methode te ontwikkelen die op een gebruiks-

vriendelijke manier toe laat om een gebruikersprofiel op te bouwen waarbij de gebruiker al controle behoudt. *De eerste feedback van de test gebruikers is positief. Het opbouwen van een filmprofiel op basis van tag-dilemma's lijkt te werken, maar het is op dit moment nog te vroeg om dit formeel te besluiten.* Verder onderzoek zal nodig zijn om een definitief besluit te nemen over deze aanpak. Indien deze aanpak uit verder onderzoek positief blijkt, kan de methode om een gebruikersprofiel op te bouwen door middel van tag-dilemma's verder uitgebreid worden naar andere sectoren zoals webshops. De resultaten van de groepsvoorspellingen zijn veel belovend. Op dit moment werden de films enkel beoordeeld op basis van de titel en de coverfoto. Een evaluatie in de toekomst zal te werk moeten gaan zoals beschreven wordt in het boek van Alexander Felfernig et al [4] waar uitgebreid beschreven wordt hoe groepsaanbevelingsalgoritmen dienen geëvalueerd te worden.

Algemeen kunnen we besluiten dat dit onderzoek een werkende proof of concept heeft ontwikkeld die de gebruiker centraal plaatst. Hiervoor werd een nieuwe methode ontworpen voor het opbouwen van een gebruikersprofiel en een groepsaanbevelingsalgoritme ontwikkeld dat werkt op basis van deze gebruikersprofielen.

Bibliografie

- [1] Poonam B.Thorat, R. Goudar en Sunita Barve. “Survey on Collaborative Filtering, Content-based Filtering and Hybrid Recommendation System”. In: *International Journal of Computer Applications* 110 (jan 2015), p. 31–36. DOI: 10.5120/19308-0760.
- [2] Sang-Min Choi, Sang-Ki Ko en Yo-Sub Han. “A movie recommendation algorithm based on genre correlations”. In: *Expert Systems with Applications* 39 (jul 2012), p. 8079–8085. DOI: 10.1016/j.eswa.2012.01.132.
- [3] Debashis Das, Laxman Sahoo en Sujoy Datta. “A Survey on Recommendation System”. In: *International Journal of Computer Applications* 160 (feb 2017), p. 6–10. DOI: 10.5120/ijca2017913081.
- [4] Alexander Felfernig e.a. “Evaluating Group Recommender Systems”. In: *Group Recommender Systems : An Introduction*. Cham: Springer International Publishing, 2018, p. 59–71. ISBN: 978-3-319-75067-5. DOI: 10.1007/978-3-319-75067-5_3. URL: https://doi.org/10.1007/978-3-319-75067-5_3.
- [5] Carlos A. Gomez-Uribe en Neil Hunt. “The Netflix Recommender System: Algorithms, Business Value, and Innovation”. In: *ACM Trans. Manage. Inf. Syst.* 6.4 (dec 2016). ISSN: 2158-656X. DOI: 10.1145/2843948. URL: <https://doi.org/10.1145/2843948>.
- [6] Dina Goren-Bar en Oded Glinansky. “FIT-recommend ing TV programs to family members”. In: *Computers Graphics* 28.2 (2004), p. 149–156. ISSN: 0097-8493. DOI: <https://doi.org/10.1016/j.cag.2003.12.003>. URL: <http://www.sciencedirect.com/science/article/pii/S0097849303002589>.
- [7] F. Maxwell Harper en Joseph A. Konstan. “The MovieLens Datasets: History and Context”. In: *ACM Trans. Interact. Intell. Syst.* 5.4 (dec 2015). ISSN: 2160-6455. DOI: 10.1145/2827872. URL: <https://doi.org/10.1145/2827872>.
- [8] Kuan Johnson. *How to Build a Simple Movie Recommender System with Tags*. Dec 2019. URL: <https://towardsdatascience.com/how-to-build-a-simple-movie-recommender-system-with-tags-b9ab5cb3b616>.

- [9] Xuan Nhat Lam e.a. “Addressing Cold-Start Problem in Recommendation Systems”. In: *Proceedings of the 2nd International Conference on Ubiquitous Information Management and Communication*. ICUIMC '08. Suwon, Korea: Association for Computing Machinery, 2008, p. 208–211. ISBN: 9781595939937. DOI: 10.1145/1352793.1352837. URL: <https://doi.org/10.1145/1352793.1352837>.
- [10] Quoc Le en Tomas Mikolov. “Distributed Representations of Sentences and Documents”. In: *Proceedings of the 31st International Conference on Machine Learning*. Red. door Eric P. Xing en Tony Jebara. Deel 32. Proceedings of Machine Learning Research 2. Beijing, China: PMLR, 22–24 Jun 2014, p. 1188–1196. URL: <http://proceedings.mlr.press/v32/le14.html>.
- [11] Hui Li e.a. “An intelligent movie recommendation system through group-level sentiment analysis in microblogs”. In: *Neurocomputing* 210 (2016). SI:Behavior Analysis In SN, p. 164–173. ISSN: 0925-2312. DOI: <https://doi.org/10.1016/j.neucom.2015.09.134>. URL: <https://www.sciencedirect.com/science/article/pii/S0925231216305872>.
- [12] Wes McKinney. “Data Structures for Statistical Computing in Python”. In: *Proceedings of the 9th Python in Science Conference*. Red. door Stéfan van der Walt en Jarrod Millman. 2010, p. 51–56.
- [13] Frederic P. Miller, Agnes F. Vandome en John McBrewster. *Internet Movie Database*. Alpha Press, 2009. ISBN: 6130099681.
- [14] Syed Mohd Ali e.a. “Movie Recommendation System using Genome Tags and Content-based Filtering”. In: nov 2017.
- [15] Michael J. Pazzani en Daniel Billsus. “Content-Based Recommendation Systems”. In: *The Adaptive Web: Methods and Strategies of Web Personalization*. Red. door Peter Brusilovsky, Alfred Kobsa en Wolfgang Nejdl. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, p. 325–341. ISBN: 978-3-540-72079-9. DOI: 10.1007/978-3-540-72079-9_10. URL: https://doi.org/10.1007/978-3-540-72079-9_10.
- [16] Maria Pera en Yiu-kai Ng. “A group recommender for movies based on content similarity and popularity”. In: *Inf. Process. Manage.* 49 (mei 2013), p. 673–687. DOI: 10.1016/j.ipm.2012.07.007.
- [17] Toon Pessemier, Simon Doooms en Luc Martens. “Comparison of Group Recommendation Algorithms”. In: *Multimedia Tools Appl.* 72.3 (okt 2014), p. 2497–2541. ISSN: 1380-7501. DOI: 10.1007/s11042-013-1563-0. URL: <https://doi.org/10.1007/s11042-013-1563-0>.
- [18] Radim Rehurek en Petr Sojka. “Gensim–python framework for vector space modelling”. In: *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic* 3.2 (2011).
- [19] Sean B Seymore. “Serendipity”. In: *NCL Rev.* 88 (2009), p. 185.
- [20] Guy Shani en Asela Gunawardana. “Evaluating Recommendation Systems”. In: deel 12. Jan 2011, p. 257–297. DOI: 10.1007/978-0-387-85820-3_8.

- [21] Douglas. Steinley. “K-means clustering: A half-century synthesis”. In: *British Journal of Mathematical and Statistical Psychology* 59.1 (2006), p. 1–34. DOI: <https://doi.org/10.1348/000711005X48266>. eprint: <https://bpspsychub.onlinelibrary.wiley.com/doi/pdf/10.1348/000711005X48266>. URL: <https://bpspsychub.onlinelibrary.wiley.com/doi/abs/10.1348/000711005X48266>.
- [22] Jesse Vig. “The Tag Genome: Encoding Community Knowledge to Support Novel Interaction”. In: *ACM Transactions on Interactive Intelligent Systems* 2 (jun 2012).
- [23] Yuan Zhang e.a. “Auralist: Introducing serendipity into music recommendation”. In: feb 2012, p. 13–22. DOI: 10.1145/2124295.2124300.

Bijlage A

Vragenlijst

Masterproef Movie Time

***Vereist**

Welkom!

In het kader van mijn masterproef onderzoek ik een manier om in groep een film te kiezen. Uw gegevens worden strikt vertrouwelijk verwerkt.

Deze vragenlijst zal max. 15 minuten in beslag nemen.
Alvast bedankt om deel te nemen aan dit onderzoek.

Indien er nog vragen/opmerkingen zijn kan u mij altijd contacteren op het volgende e-mailadres: mathies.gielen@student.uhasselt.be.

Mathies Gielen

1. *

Vink alle toepasselijke opties aan.

Ik ga akkoord met het verzamelen, verwerken en gebruiken van deze gegevens voor het beschreven onderzoeksdoel.

Inleiding

Deze thesis onderzoekt een platform om het film-keuzeproces in groep te vergemakkelijken. Deze vragenlijst peilt naar jouw individuele mening als naar de mening van de groep. In deze vragenlijst wordt aangegeven welke vragen u individueel invult en welke in groep. Gelieve voor de groepsvragen fysiek of online te overleggen met elkaar en hetzelfde antwoord in te vullen.

Kijkgedrag Individueel

De komende vragen peilen naar je kijkgedrag wanneer je alleen naar een film kijkt.

2. Hoe frequent bekijk je individueel een film? *

Markeer slechts één ovaal.

- Nooit
- Dagelijks
- Enkele keren per week
- Wekelijks
- Maandelijks
- Minder dan 1 keer per maand

3. Hoe ga je op zoek naar een film om te bekijken? *

Markeer slechts één ovaal.

- Ik kijk films die uitgezonden worden op TV
- Via film websites zoals IMDB
- Via een streamings-platform zoals Netflix
- Tips van vrienden
- Anders: _____

4. Wanneer ik individueel naar een film kijk, is het moeilijk om een gepaste film te vinden. *

Markeer slechts één ovaal.

	1	2	3	4	5	
Helemaal mee eens	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Helemaal niet mee eens

5. Ik weet welk film-genre ik wil bekijken. *

Markeer slechts één ovaal.

	1	2	3	4	5	
Helemaal mee eens	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Helemaal niet mee eens

6. Het duurt niet lang voor ik een film gevonden heb die ik wil bekijken. *

Markeer slechts één ovaal.

	1	2	3	4	5	
Helemaal mee eens	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Helemaal niet mee eens

7. Ik kan eenvoudig beschrijven wat mijn filmsmaak is. *

Markeer slechts één ovaal.

	1	2	3	4	5	
Helemaal mee eens	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Helemaal niet mee eens

8. Hoe zou je je filmsmaak omschrijven? *

9. Ik maak de keuze voor een film op basis van: *

Vink alle toepasselijke opties aan.

- de score
- de trailer
- reviews
- het genre
- acteurs
- de korte inhoud
- de cover

Anders: _____

10. Wanneer ik een film bekijk *

Markeer slechts één ovaal.

1 2 3 4 5

Ben ik meestal tevreden van mijn keuze Ben ik meestal ontevreden van mijn keuze

Kijkgedrag in groep

De komende vragen peilen naar je kijkgedrag wanneer je in groep naar een film kijkt.

11. Hoe frequent bekijk je een film in groep? *

Markeer slechts één ovaal.

- Nooit
- Dagelijks
- Enkele keren per week
- Wekelijks
- Maandelijks
- Minder dan 1 keer per maand

17. Het duurt niet lang voor we als groep een film gevonden hebben. *

Markeer slechts één ovaal.

1 2 3 4 5

Helemaal mee eens Helemaal niet mee eens

18. We kunnen als groep eenvoudig bepalen wat onze gemeenschappelijke filmsmaak is. *

Markeer slechts één ovaal.

1 2 3 4 5

Helemaal mee eens Helemaal niet mee eens

19. Wanneer we in groep een film kijken slagen we er in om een goede film te kiezen. *

Markeer slechts één ovaal.

1 2 3 4 5

Helemaal mee eens Helemaal niet mee eens

Movie Time

Movie Time is een platform om de filmkeuze als groep te vergemakkelijken. Alvorens je verder gaat maak je een account aan op het platform.
<https://www.movietime.be/accounts/signup/>

Nadat je een account hebt aangemaakt, krijg je het onderstaande scherm te zien:

The screenshot shows the 'MOVIE TIME' web application interface. At the top right, the user is logged in as 'PaulineB' with a 'Logout' button. The main content area is divided into three sections: 'Add a group' with a 'Group name' input field and a 'Create group' button; 'Join a group' with an 'Invite Code' input field and a 'Join Group' button; and 'Your groups' which lists a group named 'Test groep' with the ID 'c510CD' and a 'Details' button.

Add a group

Een van jullie maakt een groep aan door jullie groepsnaam in te vullen bij 'add a group'. Deze deelnemer zit nu automatisch in de groep. Hierna kan de rest van de groep de invite code invullen bij 'join a group'.

Vul de onderstaande vragen in zodat we de antwoorden op dit formulier kunnen linken met het juiste account.

20. Gebruikersnaam *

21. Groepsnaam *

Uw filmsmaak

Je komt nu op de detail pagina van jouw groep terecht.
Druk hier op de knop: 'Find movie taste'

Je gaat een reeks vragen krijgen. Via deze vragen krijgen we een inzicht in jouw filmsmaak. Het is dus de bedoeling dat je de vragen/dilemma's zo eerlijk mogelijk invult met wat voor jou van toepassing is.

Wanneer je de volledige reeks vragen hebt ingevuld, kom je terug op de groep pagina terecht en vul je de onderstaande vragen in.

Indien je meer info over een film wenst kan je op de film klikken om de IMDB pagina te openen

p.s.: Soms kan het even duren voor de volgende vraag verschijnt

22. Het aanmaken van het movie-profiel is gemakkelijk in gebruik.

Markeer slechts één ovaal.

1 2 3 4 5

Helemaal mee eens Helemaal niet mee eens

23. Selecteren van tags is een gemakkelijke manier om een filmsmaak te benaderen. *

Markeer slechts één ovaal.

1 2 3 4 5

Helemaal mee eens Helemaal niet mee eens

24. De Tags onderaan de groeppagina is een goede weergave van mijn filmsmaak. *

Markeer slechts één ovaal.

	1	2	3	4	5	
Helemaal mee eens	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Helemaal niet mee eens

25. De films die bij 'Your movies' staan sluiten goed aan bij mijn filmsmaak. *

Markeer slechts één ovaal.

	1	2	3	4	5	
Helemaal mee eens	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Helemaal niet mee eens

26. De films die bij 'Your movies' staan zou ik zelf bekijken. *

Markeer slechts één ovaal.

	1	2	3	4	5	
Helemaal mee eens	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Helemaal niet mee eens

27. Hoe tevreden ben je met de voorspelde films? (score op 10) *

Markeer slechts één ovaal.

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

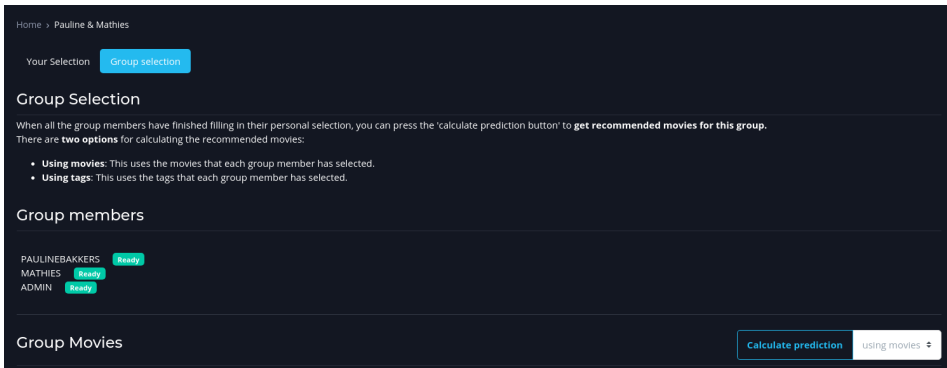
Groepsvoorspelling

De volgende vragen gaan over de films die voor jullie groep voorspeld werden.

Hiervoor is het belangrijk dat jullie als groep onderling kunnen communiceren.

Wanneer al de groepsleden de vorige stappen van het onderzoek voltooid hebben, mag je de pagina herladen en naar de tab 'Group selection' gaan.

Je zou de volgende pagina moeten zien:



Vervolgens klikt een van de groepsleden op 'Calculate prediction using movies'. De andere groepsleden kunnen hun pagina herladen wanneer de voorspelde films verschijnen om de films ook te kunnen bekijken.

De volgende vragen gaan over de films die voorspeld werden voor jullie groep.

ps.: Indien je meer info over een film wenst kan je op de film klikken om de IMDB pagina te openen

28. De lijst bevat films die ik ook zou kijken als ik alleen was. *

Markeer slechts één ovaal.

1 2 3 4 5

Helemaal mee eens Helemaal niet mee eens

29. Geef individueel een score op de aanbevolen films. (score op 10) *

Markeer slechts één ovaal.

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

30. Bespreek als groep de aanbevolen films, zit er een film tussen die jullie samen zouden bekijken? *

Markeer slechts één ovaal.

- Ja
- Nee
- Misschien

31. Geef als groep een score op de aanbevolen films (score op 10) *

Markeer slechts één ovaal.

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

Groepsvoorspelling

Er zijn twee verschillende algoritmen beschikbaar voor het aanbevelen van films. Om te weten te komen welke van de twee het beste werkt, herhalen we dezelfde vragen voor het andere algoritme.

Een van de groepsleden verandert het veld van 'using movies' naar 'using tags' en klikt vervolgens op 'Calculate prediction'. Wanneer de films zijn ingeladen kunnen de andere groepsleden hun pagina herladen om de films ook te zien.

De volgende vragen gaan over de films die voorspeld werden voor jullie groep.

32. De lijst bevat films die ik ook zou kijken als ik alleen was. *

Markeer slechts één ovaal.

	1	2	3	4	5	
Helemaal mee eens	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Helemaal niet mee eens

33. Geef individueel een score op de aanbevolen films. (score op 10) *

Markeer slechts één ovaal.

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

34. Bespreek als groep de aanbevolen films, zit er een film tussen die jullie samen zouden bekijken? *

Markeer slechts één ovaal.

- Ja
- Nee
- Misschien

39. Heb je nog overige feedback?

Einde

Bedankt om deel te nemen aan dit onderzoek.

Om er voor te zorgen dat de data op de website en deze enquête overeenkomen willen we vragen om na het beëindigen van deze vragenlijst geen wijzigingen meer te doen op de website.

Indien er nog vragen/opmerkingen zijn kan u mij altijd contacteren op het volgende e-mailadres:
mathies.gielen@student.uhasselt.be.

Deze content is niet gemaakt of goedgekeurd door Google.

Google Formulier