

Horák Krisztina

BAYES ALAPÚ AJÁNLÓRENDSZER FEJLESZTÉSE

Témavezető: Dr. Abonyi János

Mérnöki Kar, Folyamatmérnöki Intézeti Tanszék

Pannon Egyetem
Intézményi Tudományos Diákköri Konferencia
2013



HORÁK KRISZTINA

INFORMATIKAI TUDOMÁNYI SZEKCIÓ

BAYES HÁLÓ ALAPÚ AJÁNLÓRENDSZER FEJLESZTÉSE

Az üzleti világban a vállalatok számára fontosabbá vált egy ügyfél hosszú távú megtartása, mint csak új ügyfelek megszerzése. Ahhoz, hogy megtartsák a vásárlót, azonosítani kell tudniuk az ügyfél igényeit. A sokféle és változatos termékek, valamint szolgáltatások miatt kialakult információ túlterhelés megoldására, illetve a dinamikusan változó fogyasztói ízlés követésére olyan módszert kell találni, amelyek e problémák mellett is hatékonyan segítik a fogyasztói preferenciák minél pontosabb megismerését. Az ajánlórendszerek olyan speciális információszűrő rendszerek, amelyek olyan tartalmat ajánlanak a felhasználónak, amely nagy valószínűséggel tetszeni fog neki. A dolgozat célja ennek megfelelően az ajánlórendszerek hatékonyságának alátámasztása, valamint felhasználása.

Két hagyományos személyre szabott ajánlási technika a tartalom alapú (CB) és a kollaboratív szűrés (CF). A tartalom alapú ajánlás esetén a felhasználó korábbi értékelései alapján történik az előrejelzés. A CB előnye hogy igen könnyű számítani, rugalmas módszer, viszont nem tudja kezelni a kölcsönös függéseket. A kollaboratív ajánláskor a predikció inputjához hasonló adatok is felhasználásra kerülnek az előrejelzés során. A CF érzékeny az adathalmaz ritka mivoltára, és az adatbázisba való új elemek bevitelére, viszont előnye hogy intuitív, könnyen értelmezhető és interpretálható. E két technikát kombinálva úgynevezett hibrid ajánlórendszert kapunk, ahol az ajánlás minősége 6-8%-kal is jobb lehet. Az ajánlórendszerek esetén bizonytalanságokkal kell számolni, a felhasználók értékrendjének heterogenitása és az értékelés esetlegessége miatt.

A Bayes hálókat (BN) széles körben és sikeresen alkalmazzák nagyfokú bizonytalanság esetén. BN-nel könnyedén reprezentálhatók ajánlórendszer modellek. E felismerés alapján célul tűztem ki, hogy az ajánlási probléma megoldására Bayes hálózati modellt alkalmazzak, és olyan algoritmusokat tanulmányozzak, melyek szükségesek a pontos becsléshez. Dolgozatomban az általam MATLAB környezetben implementált ajánlórendszerek hatékonyságát vizsgáltam, összehasonlítottam a klasszikus technikákon alapuló rendszerek hatékonyságát a Bayes alapú rendszerekkel. A legpontosabb előrebecslést adó ajánlórendszert kerestem. A hatékonyság vizsgálatának alapját az átlagos abszolút eltérés adta. A modellek implementálásához a Minnesotai Egyetem GroupLens Research projektjében gyűjtött MovieLens filmajánló rendszerhez tartozó százezer értékelésből álló adathalmazát használtam fel.

Megállapítottam, hogy a Bayes alapú ajánlórendszerek pontosabb becslést adnak az alap technikára épülő rendszerekénél. Ezen belül pedig a hibrid módszer valóban jobbnak bizonyult a CF-nél és a CB-nél is.

Kulcsszavak: ajánlórendszerek, Bayes háló, kollaboratív szűrés, tartalom alapú szűrés

TARTALOMJEGYZÉK

Bevezetés.....	3
1. Elméleti háttér	7
1.1 Ajánlórendszerek koncepciója.....	7
1.2 Az ajánlási probléma matematikai formalizálása	9
1.3 Hasonlóság mértékek ajánlórendszerekben.....	12
1.4 Kiértékelési módszertanok	15
2. Hibrid ajánlórendszer fejlesztése.....	17
2.1 Bayes háló	18
2.2 Topológia, függőségi kapcsolatok MovieLens adathalmaz esetén.....	21
2.3 Súlyok számítása	23
2.4 Implementálás MATLAB-ban.....	28
3. Kísérlet, alkalmazás.....	32
3.1 Probléma ismertetése.....	32
3.2 Az algoritmus skálázhatóságának vizsgálata.....	33
3.3 Általános és Bayes alapú technikák összehasonlítása	34
3.4 Tartalom alapú és kollaboratív ajánlás korrelációja	37
3.5 Hibridizáció	38
3.6 Kísérlet eredményei.....	40
4. Összegzés, jövőkép	43
Függelék	45
Változók a programban	45
Fájlok-függvények a programhoz.....	46
Irodalomjegyzék.....	47

Bevezetés

Dolgozatomban ajánlórendszerek fejlesztésével foglalkozom. Ezek olyan speciális információsűrő rendszerek, amelyek felhasználói és termékprofilokat építenek tanuló algoritmusok segítségével, majd a modellek alapján ajánlanak olyan tartalmat (film, tv, zene, könyv, hír, kép, weboldal, cikk, stb.) a felhasználónak, amely nagy valószínűséggel érdekes lesz számára. (Tikk) Az ajánlórendszerek ezen felül alternatívát kínálnak új termékek és tartalmak felfedezésére is a keresés mellett. (Tikk) A keresés direkt és irányított. Ahhoz, hogy megfelelően, és személyre szabottan lehessen működtetni az ajánlórendszert a felhasználónak valamilyen elképzelése kell legyen arról, hogy mit szeretne (Tikk). Az ajánlórendszer olyan érdekes tartalmakat is felkínál, amelyeket kereséssel nem érne el a felhasználó, mert kívül esik látó- és keresési körén, mégis megfelel a saját preferenciájának. A Gravity R&D honlapján találóan és tömören megfogalmazták az ajánlórendszerek által nyújtott szolgáltatás lényegét: “Rendszerünk úgy működik, mintha egy hagyományos áruházban a vásárlókat jól ismerő eladók mindenki számára máshogy rendeznék el a kirakatot.” (Gravity, 2012). Az ajánlórendszerek modelljénél a rendszerben vannak felhasználók, akiknek vannak preferenciáik; tételek, amelyeket lehet ajánlani; valamint szavazatok, amiket a felhasználók adtak a saját preferenciájuk alapján a rendszerben lévő termékekre, szolgáltatásokra. Ezek alapján a modell elemei összekapcsolódnak. Az a cél, hogy megtudjuk melyik felhasználó mit szeret, vagy mely termékek tartoznak össze. (Konstan, 2013) Az ajánlórendszerek gazdasági jelentőséggel bírnak. Az üzleti világ folyamatosan változóban van. Maguk az ügyfelek is változnak, már nem ragaszkodnak adott márkához, könnyedén váltanak. Azok a vállalatok fognak a gazdasági versengésből győztesen kikerülni, akik a jövedelmezőség szempontjából a legígéretesebb ügyfeleket tudják megszerezni, valamint képesek lesznek őket megtartani (Abonyi, 2006). Ezért az elmúlt években egyre inkább elterjedt a vásárló központú marketing. A hagyományos termék-központú marketingben a cél a maximális bevétel az adott tranzakcióból, és nincs a kapcsolatnak távlati célja. Ezzel szemben az ügyfélkapcsolat-kezelés (Customer Relationship Management - CRM) arra a filozófiára épít, hogy sokkal inkább kifizetődőbb egy ügyfél megtartása, mint csak új ügyfelek megszerzése (Liebowitz, 2008). A CRM nagyon fontossá vált, mégisincs egy konkrét definíciója. Az ügyfélkapcsolat menedzsment az ügyfeleket középpontba helyező vállalati stratégia, amely az ügyfelek teljes körű megismerésén alapul. A CRM-et alkalmazó vállalatok ügyfélkapcsolataik értékét igyekeznek optimalizálni. A CRM rendszereket többnyire azért vezetik be, hogy a következő előnyök közül egyet vagy többet elérjenek: minőségjavítás és hatékonyságnövelés, a

termékegységre jutó fajlagos költségek csökkentése, döntéstámogatás, nagyvállalati folyamatok támogatása, vevő központú folyamatok, nyereség növelése. Az ügyfélkapcsolat-kezelés egy négylépcsős modell, melyre úgy tekinthetünk, mint egy zárt ciklusú vevőrendszer. (Peppers & Rogers, 2004) A modell négy lépését az alábbi négy kifejezés jól reprezentálja: „Ismerd meg a piacaiddat és az ügyfeleidet!”, „Célozz meg szegmenseket és egyéneket!”, „Értékesíts!”, „Szolgáltass!”. Az első lépés célja, hogy meghatározza a meglévő és a lehetséges vásárolókat, ezzel egy gazdag vevőbázist épít. A második lépésben a vásárlókat csoportosítja szükségleteik és a vállalatnak jelentett értékek alapján. Egy adott csoportba hasonló magatartást mutató ügyfelek tartoznak. Harmadik lépésben az előzetes információk alapján megtörténhet az értékesítés. A csoportosításnak köszönhetően tudhatjuk, hogy melyik ügyfélnek mit lehet eladni. Végül pedig a negyedik lépés felel az ügyfelek megtartásáért, és akár szoros kapcsolat kialakításáért az ügyfelekkel. A kommunikáció fontos eszköze a CRM-nek.

Ahhoz, hogy megtartsuk a vásárolókat, azonosítani kell tudnunk az igényeiket. A sokféle és változatos termék valamint szolgáltatások miatt a fogyasztóknak temérdek lehetőség közül kell választani, így információ túlterhelés jön létre. Ennek ellenére továbbra is ezek közül olyan információt, szolgáltatást vagy terméket kell nyújtani a felhasználónak, ami nagy valószínűséggel tetszik neki. Ennek a problémának megoldására, hogy a temérdek termék közül is ki tudjuk választani a leginkább megfelelőt illetve a dinamikusán változó felhasználói preferencia követésének igénye miatt jöttek létre az ajánlórendszerek. Az ajánlórendszereknél információsűrűséget és információ visszakeresést is alkalmazni szoktak. (Konstan, 2013) Információsűrűségről akkor beszélünk, mikor egy új, az adott felhasználó által nem ismert tartalomról állapíthatjuk meg, hogy az mennyire is fog tetszeni a felhasználóknak. (Konstan, 2013) Ezzel szemben az úgynevezett információ visszakeresés esetében egy nagy adathalmaz áll rendelkezésünkre, ahol maguk a tartalmak statikusak. (Konstan, 2013) Ekkor a tartalmat indexeljük, így könnyen visszakereshetünk egy meglévő terméket, szolgáltatást, melyekről el tudjuk dönteni, hogy az mennyire is számít a felhasználó preferenciájába. Az ajánlórendszerek célja hogy a rendszerben lévő felhasználók preferenciája alapján ajánljon egy bizonyos tartalmat, vagy megbecsülje, hogy bizonyos tartalmak mennyire tetszhetnek a felhasználónak. Annak a megbecslése, hogy egy elemet mennyire fog szeretni a felhasználó előrejelzésnek nevezik. Ez gyakran skálázható úgy, hogy megfelelően bizonyos értékelési skálán, például a csillagok száma, hányan nézték meg, hányan értékelték pozitívan. Előnye, hogy segít számszerűsíteni az elemeket, például ha valamire 5 csillagot adtunk az jó, ha 3-mat

az átlagos. Hátránya, hogy nem mindig ad helyes megoldást. Az összesített preferencia megjelenítése egy előrejelzés. Ajánlásról akkor beszélünk, amikor bizonyos számú elemet kínálunk a felhasználónak, ami valószínűleg tetszeni fog neki. Előnye, hogy legtöbbször jó választási lehetőségeket nyújt, amiből a felhasználónak nagy valószínűséggel fog tetszeni egy, kettő, vagy akár mind. Hátránya, hogy ha rosszul definiáljuk, hogy mi jó a felhasználónak, akkor nem fog a felhasználó számára megfelelő terméket, szolgáltatást találni. (Konstan, 2013) A különböző rangsorolások ajánlásoknak számítanak. Helytelen előrejelzés és ajánlás esetén is a felhasználó elveszítheti bizalmát a rendszerrel szemben, ezáltal elveszíthetjük a felhasználót, így a CRM alapfilozófiájával találnánk szembe magunkat.

Ajánlórendszer elemzésének nyolc dimenziója van (Konstan, 2013): terület, cél, kontextus, kinek a véleménye alapján történik az ajánlás, testreszabás, adatvédelem és megbízhatóság, interfészek, valamint az ajánlási technikák. Az ajánlás területe alatt azt értjük, hogy mit is akarunk kínálni, ajánlani. Az ajánlás célja az, hogy miért is történik az ajánlás. Ez lehet értékesítés vagy informálás, de akár oktatási cél a felhasználó vagy a fogyasztó számára, de még egy közösség felépítése is a felhasználók, vásárolók köréből, termék vagy tartalom alapján. Az ajánlás kontextusa alatt azt figyeljük, hogy mit is csinál a felhasználó az ajánlás alatt, például vásárol, zenét hallgat. Következő lépésben azt vizsgálhatjuk, hogy kinek vagy kiknek a véleménye alapján történik maga az ajánlás. Ez lehet akár szakértőké, vagy bármelyik emberé, vagy akár csak a vizsgált személyhez hasonló embereké. Testre szabás alatt azt nézzük, hogy az ajánlás személyre szabott, vagy sem. Nem személyre szabott ajánlásról beszélhetünk, ha csak egyes csoportoknak ajánlunk, például csak nőknek. Az ajánlások ezen felül lehetnek efemer ajánlatok, amelyek csak egy rövid ideig vannak jelen, valamint lehetnek tartós ajánlások is, amik a felhasználó számára hosszú távú ajánlást kívánnak tenni. Ajánlórendszerek esetén is fontos a személyes adatok, személyazonosság védelme. Az elemzés hetedik dimenziója az interfészek vizsgálata. Ezek lehetnek output és input interfészek. Kimenő például az előrejelzés, ajánlás, szűrés, bemenő pedig például az értékelések, szavazatok, vásárlás, kattintások. Ajánlási algoritmusok közül a nem személyre szabott összegző statisztikák, a tartalom alapú szűrés (CB), kollaboratív szűrés (CF) valamint egyéb ajánlási algoritmusok a leggyakoribbak. (Smita & Akshita, 2013) A nem személyre szabott összegző statisztikának külső forrásból származnak az adatai, például best-seller-ek, a legnépszerűbb trendek. Ez az algoritmus még közösségi értékelések összegzéséből is dolgozik, például a legjobban kedvelt elemek. Nem veszi figyelembe a felhasználók egyéni preferenciáit. A termék-egyesülésen alapuló ajánlások is igen hasznosak lehetnek, de ehhez

meg kell határozni az adatok közötti összefüggéseket, az adatok forrását, hatályát. A nem-személyre szabott ajánlások kihívásokkal telítettek a változatos lakossági ízlések miatt, mégis igen gyakran előfordul. A nem-személyre szabott ajánlások is lehetnek hatékonyak, ha megfelelően alkalmazzuk őket. Meg kell érteni a kapcsolatot az átlag és a felhasználók szükségletei között, fontosak a helyes átlagértékek. Egyéb ajánlási algoritmus lehet interaktív ajánlás kritika vagy interjúk alapján, illetve a különböző technikák alapján hibrid módszerek is alkalmazhatók. A tartalom alapú ajánlás esetén a felhasználó korábbi értékelései alapján, vagy a felhasználó saját elmondása alapján felhasználói profiloldal épül fel, amely alapján történik később a predikció.(Engedy, 2010) A kollaboratív ajánláskora rendszerben lévő értékelések, vélemények, tapasztalatok kerülnek felhasználásra, majd azon meglévő információk segítségével történik az előrejelzés. A kollaboratív szűrés és a tartalom alapú ajánlás technikák kombinálhatóak, ekkor úgynevezett hibrid ajánlórendszert kapunk, ahol az ajánlás minősége 6-8% is javulhat az előző kettő algoritmushoz képest (Engedy, 2010).

Az ajánlórendszerek esetén bizonytalanságokkal kell számolni a felhasználók értékrendjének heterogenitása és az értékelés esetlegessége miatt. A Bayes hálókat (BN) széles körben sikeresen alkalmazzák nagyfokú bizonytalanság esetén. (Russell & Norvig, 2005) BN-nel könnyedén reprezentálható kollaboratív, tartalom alapú és hibrid ajánlórendszer modell is, mely a felhasználó preferenciáit jól előrejelzi, ezen felül átlátható, könnyedén módosítható modell. Ezen felismerés alapján célul tűztem ki a Bayes háló alapú modellek hatékonyságának vizsgálatát ajánlási probléma megoldására, ezek összehasonlítását az általános ajánlási technikákkal, és olyan algoritmusokat tanulmányozását, melyek szükségesek a pontos becsléshez. A legpontosabb előrebecslést adó ajánlórendszert kerestem. A hatékonyság vizsgálatának alapját az átlagos abszolút eltérés adta. A modellek implementálásához a Minnesotai Egyetem GroupLens Research projektjében gyűjtött MovieLens filmajánló rendszerhez tartozó százezer értékelésből álló adathalmazát használtam fel (GroupLens, 2003). Ezen adatsorban közel ezer felhasználó értékelését ismerhetjük meg a több mint ezerhatszáz film esetén. A különböző ajánlási algoritmussal felépített modelleket felhasználva előlegeztem meg a hiányzó értékeléseket.

Dolgozatom 1. fejezetében a témakörhöz tartozó elméleti háttérrel fogom biztosítani az ajánlórendszerekről, valamint a hasonlóság mértékekről és a kiértékelési módszertanokról. A 2. fejezetben a Bayes háló ismertetése és a hibrid ajánlórendszerek megvalósítása fog szerepelni, illetve ennek az ajánlórendszer implementálásának folyamata MATLAB

környezetben. Ezt követően a 3. fejezetben szerepel maga a kísérleti rész, melyben az ajánlórendszerek felhasználásáról, vizsgálatáról és kiértékeléséről lesz szó. Végül az 4. fejezetbe egy rövid összefoglalót írtam. A Függelékben található a saját ajánlórendszeremben használt változók, függvények.

1. Elméleti háttér

E fejezetben mutatom be az ajánlórendszerek koncepcióját, az ajánlási probléma matematikai formalizálását. Áttekintem a rendszer működésének egyik legfontosabb elvét, a hasonlósági mértékeket, illetve a rendszer fejlesztéséhez azokat a mérőszámokat, amelyek alapján minősíteni tudom a fejlesztett ajánlórendszerem.

1.1 Ajánlórendszerek koncepciója

Ahhoz hogy az ajánlórendszer használható legyen felhasználói és termék profilok felépítésére van szükség. A profilokhoz szükséges adatokhoz hozzá lehet jutni explicit és implicit értékelések által (Konstan, 2013). Az explicit értékelés alapötlete, hogy ha meg akarjuk tudni mit is gondol a felhasználó, akkor meg kell tőle kérdezni. Az értékelést különböző alkalmakkal lehet kikérni: a fogyasztás alatt, vagy közvetlenül a fogyasztás után; egy bizonyos idő elteltével; illetve csak egy elvárt szavazat is kérhető, ekkor nem volt fogyasztás, nincs az adott termékkel tapasztalata a felhasználónak. Az értékeléssel is vannak nehézségek, például felmerül a kérdés, hogy a besorolás mennyire pontos, mennyire megbízható. Előfordul hogy a felhasználó preferenciája változik, ekkor biztosítani kell a felhasználói profil frissülését. Az implicit adatokat a felhasználói tevékenységek alapján gyűjthetjük össze. Alapvető különbség az explicit kategóriával szemben, hogy az implicit nem fejezi ki a felhasználó preferenciáját. A felhasználó tevékenységeiből sok, sokatmondó adat nyerhető. Adatot nyerhetünk abból is ha megfigyeljük hogy a felhasználó mennyi időt tölt olvasással, hallgatással, vizsgálhatjuk a felhasználó figyelmének irányulását.

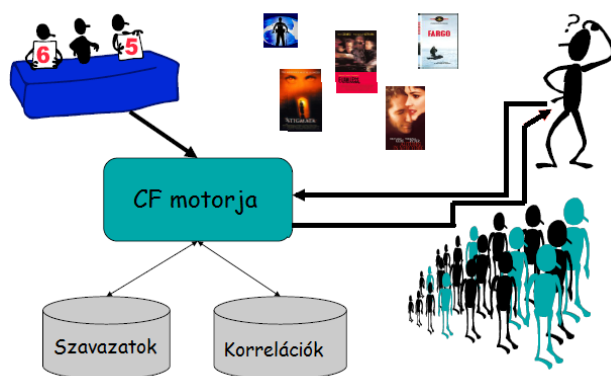
Az általam vizsgált két ajánlási módszer a tartalom alapú és a kollaboratív szűrés alapú ajánlás. A tartalom alapú ajánlás esetén a felhasználó korábbi értékelései alapján, vagy a felhasználó saját elmondása alapján felhasználói profil oldal épül fel, amely alapján történik később a predikció. (Campos, Fernández-Luna, Huete, & Rueda-Morales, 2010) Ehhez az ajánlani kívánt termékek, szolgáltatások is rendelkeznek termékprofillal, melyben a tulajdonságaik szerepelnek. A tartalom alapú előrejelzés előnye hogy igen könnyű számítani,

független a többi felhasználótól így egyetlen felhasználó esetén is életképes a rendszer, rugalmas módszer, a ritkaság nem okoz gondot. Viszont nem tudja kezelni a kölcsönös függéseket. Elképzelhető, hogy nem mindig a már magasra értékelthez leginkább hasonló elem ajánlása a legideálisabb. Illetve ha a felhasználó tudja változtatni a profilját, akkor a régi adatokat, preferenciákat is rendszeresen ellenőrizni kell, mivel folyamatosan változhatnak a régi adatok is. (Dietmar, 2011) A kollaboratív szűrés alapgondolata, hogy azok a felhasználók, akik a múltban hasonló dolgokat kedveltek a jövőben is hasonlóan fognak viselkedni. (Campos, Fernández-Luna, Huete, & Rueda-Morales, 2010) A rendszerben lévő értékelések, vélemények, tapasztalatok kerülnek felhasználásra, majd azon meglévő információk segítségével történik a predikció. CF alkalmazására három technika létezik: felhasználó-, termék-, illetve dimenziócsökkentés alapú (Campos, Fernández-Luna, Huete, & Rueda-Morales, 2010). Felhasználó alapú kollaboratív szűrésen a hasonló felhasználók alapján történik az előrejelzés. Ehhez hasonlóan a tartalom alapú CF szűrés esetén az termékek hasonlósága alapján próbáljuk megmondani a jövőbeli pontszámot az adott termékekre. Dimenziócsökkentéskor a meglévő mátrix mérete csökken úgy, hogy az továbbra is tükrözze a felhasználó preferenciát, nem veszhet el lényegi információ. A kollaboratív szűrés érzékeny az adathalmaz ritka mivoltára, és az adatbázisba való új elemek bevitelére, valamint ha a közösen értékelt elemek száma alacsony az nagyon torzíthat. Előnye hogy intuitív, könnyen értelmezhető és interpretálható.(Dietmar, 2011)

A kollaboratív alapú szűrési technikák megadott lépéseket használnak (Engedy, 2010). Először a cél felhasználónak preferencia vektorát kell megismerni. Ezután a többi felhasználó közül meg kell találni a cél felhasználóhoz leginkább hasonló felhasználó csoportot. Ezek a hasonló felhasználók által megvásárolt, használt termékei megismerhetők, ezeknek a termékeknek a várható értékelését adjuk meg a cél felhasználó számára. Ennek alapján történik az ajánlás vagy az előrejelzés.

Klasszikus ajánlórendszer a MovieLens filmajánló portál¹, melyet én is vizsgálok később a dolgozatomban. A MovieLens adatbázisában különböző filmek, felhasználók szerepelnek, a rendszer pedig próbál olyan filmet ajánlani az egyes felhasználóknak, ami számukra nagy valószínűséggel tetszeni fog (Konstan, 2013). Működését az 1.1. ábra jól szemlélteti.

¹ <http://movielens.org/>



1.1. ábra Példa CF működésre

Ha egy felhasználó megnéz egy filmet, akkor azt pontozza, a pontszám pedig bekerül a rendszer motorjába, ami egy szűrő egység. Minden szavazat eltárolódik az adatbázisban. A szavazatok alapján páronkénti korreláció számítás segítségével a rendszer eltárolja az adott felhasználóhoz hasonló másik felhasználókat. Így a rendszer a hasonlóságok alapján ajánl filmet az aktuális felhasználónak (Konstan, 2013).

Tartalom alapú előrejelzés esetén pedig a célfelhasználó korábbi szavazatai számítanak, a többi felhasználótól függetlenül történik az előrejelzés, ajánlás. A célfelhasználó által legjobbra értékelt filmekhez hasonló filmeket fog ajánlani a rendszer.

1.2 Az ajánlási probléma matematikai formalizálása

Általánosan nm darab termék (I), nf darab tulajdonság (F) és nu db felhasználó (U) van a rendszerben. A felhasználók valamilyen módon kapcsolatba kerülnek a termékkel, például megvásárolták, megnézték. A termékeket a felhasználók pontozzák. (Campos, Fernández-Luna, & Huete, 2005)

Elsőként bevezethető egy $nm \times nf$ –es D mátrix, ami megadja, hogy mely termék mely tulajdonsággal vagy tulajdonságokkal jellemezhető. D egy bináris mátrix, ahol $d_{i,j} = 0$ ha az i . filmre a j . tulajdonság nem jellemző, $d_{i,j} = 1$, ha igen. Az alábbi 1.1. táblázatban megadott D mátrixból leolvasható, hogy az első termékre a második és a harmadik tulajdonság jellemzőbe sorolták be. A második termékre az első és a harmadik tulajdonság, míg a harmadikra csak a második tulajdonság jellemző.

1.1. táblázat Tulajdonsági mátrix részlet

	F₁	F₂	F₃	...
I₁	0	1	1	
I₂	1	0	1	
I₃	0	1	0	
...				

Bevezethető még egy $nu \times nm$ –es **R** segédmátrix is, ami pedig azt mutatja, hogy mely felhasználó mely terméket értékelte. Ez egy szintén bináris mátrix, ahol $r_{i,j} = 0$ ha az *i.* felhasználó a *j.* terméket nem értékelte, $r_{i,j} = 1$, ha igen. Az 1.2. táblázat azt mutatja, hogy az első felhasználó a második és a harmadik terméket megnézte/megvette, pontozta, az elsőt pedig nem, a második felhasználó már mindegyiket értékelte, míg a harmadiknak csak a második termékkel volt már kapcsolata.

1.2. táblázat Értékelés meglétét szemléltető mátrix részlet

	I₁	I₂	I₃	...
U₁	0	1	1	
U₂	1	1	1	
U₃	0	1	0	
...				

Magukat a pontszámokat egy **R**-rel megegyező nagyságú **Y** mátrix tárolja. Itt már magukat az értékelési eredmények jelennek meg. Értelem szerűen ahol az **R** mátrixban 1 volt, ott az **Y** mátrixban 1-től 5-ig kerülnek az értékek a szavazati pontszám értelmében. Az 1.3. táblázatot megvizsgálva látható, hogy az első felhasználó a második terméket ötösre értékelte, míg a harmadik tétel már kevésbé tetszett neki, így egyest adott rá. A második felhasználó az első két termékre egyest adott, míg a harmadikra hármast. A harmadik szereplőnek az általa látott/vásárolt második tétel hármast értékelési pontot ért.

1.3. táblázat Értékelések mátrixának részlet

	I_1	I_2	I_3	...
U_1	0	5	1	
U_2	1	1	3	
U_3	0	3	0	
...				

Mivel az ajánlórendszerek alapja a hasonlóság, így bevezettem egy $\mathbf{H}$ mátrixot, ami $nu \times nu$ –es mátrix, a felhasználók közötti hasonlóság mértékeket tárolja. A fejezet későbbi részében ismertetett módon kiszámítható a hasonlóság mértéke. A $\mathbf{H}$ értékei a $[0,1]$ intervallumból kerülnek ki. Minden film önmagával maximálisan hasonló. Maga a hasonlóság megléte szimmetrikus, de a hasonlóság mértéke eltérő lehet. Az 1.4. táblázat esetén látható például hogy az első felhasználó hasonlósága a másodikkal 1, míg a második felhasználó az elsővel csak 0.6.

1.4. táblázat Felhasználók hasonlósági mátrix részlet

	U_1	U_2	U_3	...
U_1	1	1	0.5	
U_2	0.6	1	0.3	
U_3	1	1	1	
...				

Ezen felül a tartalmak között is definiálható a hasonlóság, melyet egy $\mathbf{S}$ mátrix tárol, ami $nm \times nm$ –es méretű. Bináris mátrix, mivel csak a hasonlóság meglétét tárolja. Ha az $s_{i,j} = 1$, ha az i . és a j . tartalom hasonló, $s_{i,j} = 0$, ha nem. A hasonlóság a tartalmak esetén is szimmetrikus, ha az i . tartalom hasonló a j . tartalomhoz, akkor a j . elem is hasonló lesz az i . elemhez. Az 1.5. táblázat mutatja, hogy a mely termék melyik másik termékkel hasonló. Két tétel akkor lesz hasonló, ha legalább egy tulajdonságban hasonlítanak. Így a 1.1. táblázatból kiolvasható, hogy az első tartalom mindegyikhez hasonló, de a második és a harmadik termék már nem hasonlóak.

1.5. táblázat Termék hasonlósági mátrix részlet

	I ₁	I ₂	I ₃	...
I ₁	1	1	1	
I ₂	1	1	0	
I ₃	1	0	1	
...				

Az $\mathbf{Y}$ mátrix nagyon hiányos. Az ajánlórendszer célja a teljes $\mathbf{Y}$ mátrix előrejelzése, így a felhasználóknak olyan tartalmat lehet ajánlani, ami nagy valószínűséggel tetszeni fog neki. Fontos, hogy az ajánlórendszerek megfelelő ajánlatokat, előrejelzéseket adjon, máskülönben a felhasználók elveszíthetik hitüket a rendszerrel kapcsolatban, ezáltal elhagyják az adott rendszert. Ahhoz, hogy meg tudjuk mondani, mennyire is volt jó a becslés különböző kiértékelési módszertanok állnak rendelkezésre.

1.3 Hasonlóság mértékek ajánlórendszerekben

A személyre szabott ajánlórendszerek alapja a hasonlóságok keresése felhasználók vagy termékek között. Ezek a rendszerek a hasonlósági mátrixokra támaszkodva alcsoportokat képeznek, felhasználók esetén azok tartoznak ugyanazon alcsoportba, akiknek az ízlése hasonló; termékek esetén pedig tulajdonságuk alapján hasonló termékek tartoznak össze. A hasonlóság fontossága miatt ebben az alfejezetben ismertetem a hasonlóság mértékeket, menet közben részletesen ismertetem a fogalmakat, jelöléseket.

Felhasználó alapú szűrés esetén gyakran a következő módszert alkalmazzák (Ricci, Rokach, Shapira, & Kantor, 2011): Legyen u_i a felhasználók halmaza ($i = 1, \dots, n_u$, ahol n_u a felhasználók száma), p_j pedig a termékek halmaza ($j = 1, \dots, n_m$, ahol n_m a termékek száma). Ekkor az $n \times m$ -es mátrixban szerepelnek a felhasználók által a termékekre adott pontszámok, v_{ij} . Ha egy terméket még nem pontozott az adott felhasználó, akkor az adott v_{ij} értéke 0. Ekkor, ha előre szeretnénk jelezni ezen ismeretlen szavazat értékét, az alábbi (1.1) képlet használható,

$$v_{ij}^* = v_i + K \sum_{k \neq 0} u_{ik} (v_{kj} - v_k) \quad (1.1)$$

ahol v_i az átlag értékelése az i felhasználónak, K egy normalizálós tényező oly módon, hogy u_{ik} összege 1, és u_{ik} az i . és k . felhasználó közötti hasonlóságot adja meg.

Gyakran használatos a minták közötti távolság mérésére a koszinusz távolság felhasználása, melyet a (1.2) képlet mutat (Ricci, Rokach, Shapira, & Kantor, 2011):

$$\cos(u_i, u_j) = \frac{\sum_{k=1}^{nm} v_{ik} v_{jk}}{\sqrt{\sum_{k=1}^{nm} v_{ik}^2 \sum_{k=1}^{nm} v_{jk}^2}} \quad (1.2)$$

ahol u_i, u_j két különböző felhasználó, v_{ik} és v_{jk} pedig az általuk adott pontok, nm pedig a termékek száma. Probléma hogy nem minden értékelés ugyanolyan súlyú: ahol mindenki egyetért az kevésbé számít, mint ahol vegyesek a vélemények. Nagy torzítást okozhat, ha a közösen értékelt elemek száma alacsony. Az adatok összefüggéseit nem fejtí meg és nagy elemszám esetén túlságosan nagy erőforrás szükséglete van, ezért kisebb adatbázisokra vagy az adatbázisokból kinyert adathalmazokra alkalmazzák.

Két mennyiség közötti kapcsolatot célszerű egy mérőszámmal leírni. (Fidy Judit dr., 2005) Az ilyen mérőszám közül a korrelációs együttható a legelterjedtebb. A korrelációs együttható értéke $[-1,1]$ között van. Az együttható értékéből lehet következtetni az elemek közötti kapcsolat mértékére. A korrelációs értékek grafikusán ábrázolhatóak, amely ábráról sokatmondó információk olvashatóak le. Ha a pontok nem fekszenek egy egyenes mentén, akkor azt mondjuk, hogy nincs korreláció közöttük (0), ha az együttható értéke közel van a nullához, akkor gyenge korreláció van. Ha a pontok egy egyenes mentén helyezkednek el, akkor a korrelációs együttható közel van -1 -hez vagy $+1$ -hez és magas a korreláció, ha pedig a pontok rajta vannak az egyenesen és az egyenes csökkenő, akkor -1 , ha növekvő, akkor 1 .

Az egyik leggyakrabban használt mérőszám a Pearson-féle korrelációs együttható. Pearson-féle korrelációs együttható számítása esetén a változóknak folytonosnak kell lenni és mindegyik normális eloszlású, vagy ha az egyik változó értékei rögzítettek, akkor a másik változó eloszlása minden rögzített változó esetében normális eloszlást kövessen. A változókat egymástól függetlenül kell mérni. Kiugró értékek esetén még nagyobb hangsúlyt kap a pontfelhő ábra tanulmányozása. Ha két változó kapcsolata magas korrelációt eredményez, megpróbálhatjuk az összefüggést egy ideális egyenessel jellemezni. A páronként korrelációs együttható az (1.3) képlet segítségével határozható meg (Hahsler, 2011),

$$\varphi_{ik} = \frac{\sum_j (v_{ij} - \bar{v}_i)(v_{kj} - \bar{v}_k)}{\sqrt{\sum_j (v_{ij} - \bar{v}_i)^2 \sum_j (v_{kj} - \bar{v}_k)^2}} \quad (1.3)$$

ahol v_i és v_k két objektum, $\bar{v}_i$ az i . objektum összes változójára számított átlagos érték, amely az (1.4) képlettel számítható ki.

$$\bar{v}_i = \frac{1}{N} \sum_{k=1} v_{i,k} \quad (1.4)$$

A hasonlóság számításakor én a Pearson-féle korrelációs együtthatót használtam a kísérletemben, némi korrekcióval. Korrekció azért szükséges, mert van olyan eset ahol a magas korreláció annak köszönhető, hogy csak nagyon kevés közös elemet értékelték a vizsgált felhasználók. Erről részletesebben a 2. fejezetben lesz szó.

Az 1.6. táblázat egy konkrét példa a hasonlóság számítására CF felhasználásakor. A példában öt felhasználó és öt termék szerepel. Ezek közül négy felhasználó minden értékelését ismerjük. Cél, hogy a hiányzó értékelést is előre jelezzük, ami Anna ötödik elemre vonatkozó pontszáma.

1.6. táblázat Példa előrejelzésre CF segítségével

	I ₁	I ₂	I ₃	I ₄	I ₅
Anna	5	3	4	4	?
U ₁	3	1	2	3	3
U ₂	4	3	4	3	5
U ₃	3	3	1	5	4
U ₄	1	5	5	2	1

Először ki kell választani Annához hasonló felhasználókat (szomszédok), akik értékelték az 5. elemet. Ezután a becült értékelés a szomszédok értékelésének átlaga alapján számítható.

(1.3) képlet segítségével számolhatóak a páronkénti hasonlóságok, de fontos megjegyezni, hogy csak azon elemek esetén, amelyet mindketten értékelték. A hasonlósági értékek a következők: $\varphi_{Anna,1} = 0.85$, $\varphi_{Anna,2} = 0.7$, $\varphi_{Anna,3} = 0$, $\varphi_{Anna,4} = -0.79$. Ezután a hasonlóságokkal súlyozva az értékeléseket kiszámítható Anna pontszáma is.

1.4 Kiértékelési módszertanok

Fontos, hogy az ajánlórendszerek megfelelő ajánlatokat, előrejelzéseket adjon, máskülönben a felhasználók elveszíthetik hitüket a rendszerrel kapcsolatban, ezáltal elhagyják az adott rendszert. Ahhoz, hogy meg tudjuk mondani, mennyire is volt jó a becslés különböző kiértékelési módszertanok állnak rendelkezésre. Ezeknél a módszertanoknál a megbecsült és a már valós értékeket hasonlítjuk össze. Ezért ezen alfejezet célja, hogy bemutassak néhány kiértékelési módszertant.

Az egyik leggyakrabban használt módszer az átlagos abszolút hiba (MAE) számítása. Kiszámítása egyszerű, a valós értékelések és a becslt értékelések eltérésének abszolút értékeinek az átlagát számolja, formálisan az (1.5) képlet mutatja be,

$$MAE = \frac{\sum_{i=1}^N |v_{ij} - \hat{v}_{ij}|}{N} \quad (1.5)$$

ahol v_{ij} a valós értékelés, $\hat{v}_{ij}$ a becslt értékelés, N pedig a vizsgált minták száma.

Négyzetes középérték hibát (RMSE) akkor érdemes használni, ha fontos, hogy kis hibával működjön a rendszer. Az RMSE akkor hasznosabb, ha a hiba nagysága is számít mivel a hibát négyzetre emelés előtt átlagoljuk, így viszonylag nagy súlyt ad a nagy hibáknak. Formálisan az (1.6) képlet írja le.

$$RMSE = \sqrt{\frac{\sum_{i=1}^N |v_{ij} - \hat{v}_{ij}|^2}{N}} \quad (1.6)$$

A MAE és az RMSE együtt is használható. Az RMSE mindig nagyobb vagy egyenlő mint a MAE. Minél nagyobb az eltérés a kettő között, annál nagyobb a variancia az egyes hibák között a mintában. Ha a kettő egyenlő, akkor az összes hiba ugyan olyan mértékű. Az átlagos abszolút hiba és a négyzetes középérték hiba értéke is nullától végtelenig terjed.

Az F-mérték egy másik gyakran használt mérőszám (Olson & and Delen, 2008). Kiszámítását az (1.7) képlet mutatja. Számoláshoz szükségünk van a fedésre illetve a pontosságra, mégpedig az F-mérték e kettő harmonikus közepe. $[0,1]$ intervallumból kerül ki F értéke, minél nagyobb, annál jobb az ajánlás.

$$F = 2 * \frac{\text{pontosság} * \text{fedés}}{\text{pontosság} + \text{fedés}} \quad (1.7)$$

A fedés az összes helyesnek vélt és valóban helyes ajánlás aránya az összes helyes ajánlással. A pontosság pedig a helyesnek vélt és valóban helyes ajánlás aránya az összes helyesnek vélt ajánlással. Ezek megadásához szükségünk van egy keveredési mátrixra, melyet az 1.7. táblázat szemléltet (Hahsler, 2011).

1.7. táblázat Keveredési mátrix

Aktuális feltétel			
Teszt eredmény	A feltétel teljesül		A feltétel nem teljesül
	Pozitív	<i>A feltétel teljesül</i>	<i>A feltétel nem teljesül</i>
	(megfelelő)	+ pozitív teszteredmény = tp	+ pozitív teszteredmény = fp
	Negatív (nem megfelelő)	<i>A feltétel teljesül</i> + negatív teszteredmény = fn	<i>A feltétel nem teljesül</i> + negatív teszteredmény = tn

Keveredési mátrix felírásához meg kell keresni azokat az ajánlásokat, amelyeket ajánlott a rendszer, pedig nem kellett volna (fp), amelyeket nem ajánlott pedig kellett volna (fn), illetve amiket helyesen ajánlott (tp). E mátrix segítségével megadható a keresett értékek: a fedés az (1.8) képlettel, a pontosság az (1.9) képlettel számolható. (Ng, 2012)

$$\text{fedés} = \frac{tp}{tp + fn} \quad (1.8)$$

$$\text{pontosság} = \frac{tp}{tp + fp} \quad (1.9)$$

A korábban említett képlettel (1.7) már kiszámítható az F-mérték. Probléma ezzel a módszerrel az, hogy nehéz kiszámítani, mivel ehhez tudnunk kell melyek relevánsak, és melyek nem. Nehéz megtudni, hogy mi a fontos a felhasználók számára. Ez a módszer inkább kis adathalmazokon használható jól. A fedést nehezebb számítani, mint a pontosságot.

Ajánlórendszerek esetén gyakori a top-N eljárás (Hahsler, 2011). Például egy filmajánló rendszer a top-N eljárásakor azt az N darab filmet ajánlja a felhasználónak, amelyek nagy valószínűséggel leginkább tetszeni fog neki az összes film közül. Ezzel az eljárásnál jól használható az előzőekben ismertetett pontosság és fedés. A becsült mátrixból és az eredeti mátrixból is meg kell adni a top-N halmazt, amelyek azokat a termékeket fogják tartalmazni, amik a legtöbb pontszámot kapták. Azok a filmek lesznek a helyesen ajánlottak, amelyek mindkét mátrix top-N-jében szerepelnek. Amelyek csak az eredetiben azok a helytelenül nem ajánlottak, amik csak a becsült mátrix top-N-jében pedig az helytelenül nem ajánlottak. Ezután a korábban ismertetett képletek segítségével megadható a fedés (1.8), pontosság (1.9) és az F-mérték (1.7) értéke is.

A keveredési mátrix alapján szintén számíthatjuk a MAE értékét, mégpedig az (1.10) képlettel (Hahsler, 2011).

$$MAE = \frac{fn + fp}{tp + fp + fn + tn} \quad (1.10)$$

A pontosságának méréséhez én a MAE mérőszámot valamint az F-mértéket használtam, melyek eredményét a 3. fejezetben részletesen ismertetek.

2. Hibrid ajánlórendszer fejlesztése

Általános ajánlási technikák közül a kollaboratív és a tartalom alapú algoritmusra fektettem a hangsúlyt, és ezek ötvözésére: a hibrid ajánlásra. Célom, hogy az alap ajánlási technikákat javítsam, ehhez Bayes hálót használtam, mivel számos területen jónak bizonyult (Russell & Norvig, 2005). Ehhez a (Campos, Fernández-Luna, Huete, & Rueda-Morales, 2010) irodalmat részleteiben feldolgoztam, az ott megismert technikákat felhasználva elkészítettem a saját Bayes alapú ajánlórendszerem. A következő alfejezetekben elsőként a Bayes hálót ismertetem, ami szükséges az ajánlórendszer topológiájának, alkalmazott szabályainak a megértéséhez, majd pedig a (Campos, Fernández-Luna, Huete, & Rueda-Morales, 2010) irodalomból származó súlyok számításáról szóló elméleti részt mutatom be. Végül pedig helyet kap, hogy hogyan terveztem meg, majd építettem ki MATLAB-ban az ajánlórendszert.

2.1 Bayes háló

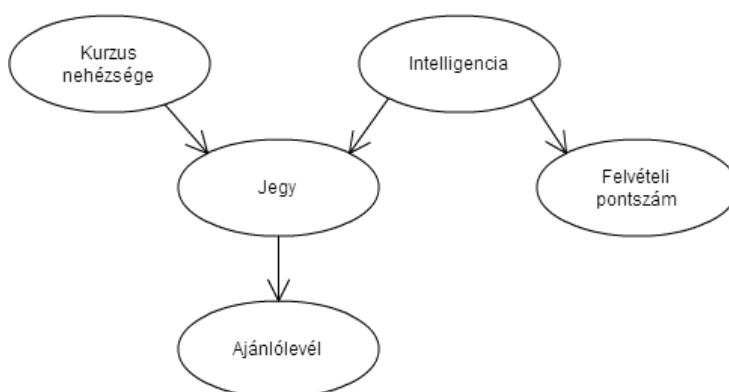
Ez az alfejezet bevezet egy Bayes-hálónak (BN) nevezett adatstruktúrát a változók közötti függőség leírásához, és bármely együttes valószínűség-eloszlás függvény tömör megadásához. A Bayes háló egy adott feladat változóinak oksági struktúráját leíró irányított körmentes gráf. (Russell & Norvig, 2005) A körmentesség azt jelenti, hogy nincs benne kör, nem keresztezik egymást az élek, nem megy vissza él onnan ahonnan elindult. Irányított élek (nyilak) egy halmazt összeköt bizonyos csomópontpárokat. Ha létezik nyíl az X csomóponttól az Y csomópontig, azt mondjuk, hogy az X szülője az Y-nak. Ennek az irányított körmentes gráfnak a csomópontjai képviselik a valószínűségi változókat $X_1, \dots, X_n$, amely változók lehetnek diszkrétnek vagy folytonosak. A gráf minden x_i csomópontjához tartozik egy feltételes valószínűségi eloszlás $P(x_i | \text{Szülők}(x_i))$, ami megmondja a szülőktől való függőségét, a szülők hatását a csomóponti változóra. Egy helyesen létrehozott hálóban az X csomópontot az Y csomóponttal összekötő nyíl intuitív jelentése rendszerint az, hogy az X-nek közvetlen befolyása van az Y-ra. A tárgyterület szakértője számára általában könnyű eldönteni, hogy milyen közvetlen befolyások teljesülnek egy adott területen – valójában sokkal könnyebb, mint a megfelelő valószínűségeket megadni. BN reprezentálja az együttes valószínűségi eloszlást a Bayes hálózat láncszabályán keresztül, melyet a (2.1) egyenlet szemléltet.

$$P(X_1, \dots, X_n) = \prod_i P(x_i | \text{Szülők}(x_i)). \quad (2.1)$$

Bayes háló felépítéséhez először szükséges megállapítani magukat a csomópontokat, majd ezek közötti függéseket. (Koller, 2012) A csomópontok az állítások, az élek pedig a köztük lévő kapcsolatokat reprezentálják. A gráf mellé még le kell írni az ok-okozati hatásokat, ezért minden csomóponthoz hozzárendelhető egy valószínűségi táblázat, melynek minden sorának összege egy. Azok a csomópontok, melyeknek nincs szülője, azok úgynevezett prior valószínűségek. Mivel a Bayes háló valószínűségi értékeket reprezentál, ezért ezek mindig nagyobbak, vagy egyenlők mint 0, mivel a valószínűségi eloszlások mindig nemnegatívak. (Koller, 2012) Ha a háló csomópontjaiban a valószínűségeket összeadjuk mindig egyet kapunk. Ha a Bayes-háló topológiája kész, már csak az egyes változókhoz tartozó feltételes valószínűségi-eloszlásokat kell meghatározni a szülőkkel, mint feltételekkel. A topológia és a

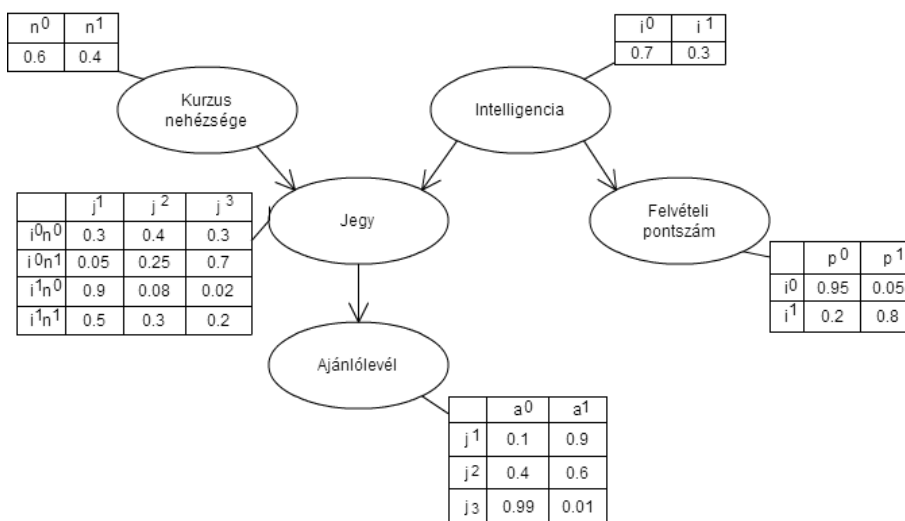
feltételes eloszlások együttese elegendő, hogy megadja (implicit módon) az összes változó feletti együttes valószínűség-eloszlás függvényét. (Koller, Friedman, Getoor, & Taskar, 2007)

A következő példában van egy tanuló, aki egy bizonyos kurzusra jár. (Koller, 2012) A példában szereplő változók a kapott jegy (**J**), a diák intelligenciája (**I**), felvételi pontszáma (**P**), a kurzus nehézsége (**N**) és a kapott ajánlólevél (**A**). Az **N**, **I**, **A**, **P** binárisak, míg a **J** három értéket vehet fel attól függően, hogy a diák hányas osztályzatot ért el. A kapott osztályzatot középre leírva feltevődik a kérdés: Mitől függ a tanuló jegye? Attól, hogy milyen nehéz a tanfolyam, illetve hogy milyen okos a diák. A felvételi pontszám csak az intelligenciától függ, az ajánlólevél pedig a kapott osztályzattól. A kapcsolatok meghatározás után felrajzolható a függőség egy modellje, melyet a 2.1. ábra szemléltet.



2.1. ábra Példa Bayes hálóra

A gráf mellé még le kell írni az ok-okozati hatásokat, ezért minden csomópontához hozzárendelünk egy valószínűségi táblázatot. (2.2. ábra)



2.2. ábra Példa Bayes hálóra valószínűségi táblázattal

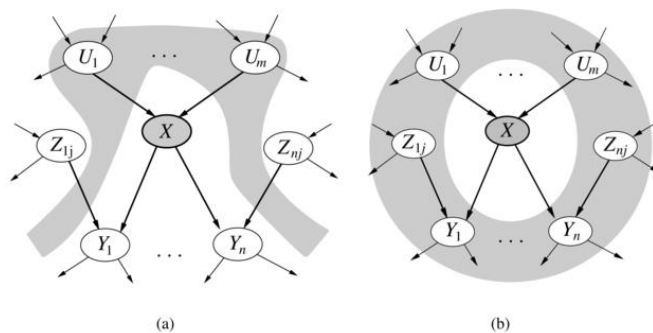
Látható hogy a táblázatok minden sorainak összege valóban egy. Ezen kívül leolvasható az is, hogy a kurzus nehézsége és a diák intelligenciája feltétel nélküli prior valószínűségek $P(N), P(I)$; míg a maradék három feltételes valószínűségekkel rendelkeznek, mivel a kapott jegy függ a diák intelligenciájától és a kurzus nehézségétől is $P(J|I, N)$; a felvételi pontszám a diák intelligenciájától függ $P(P|I)$; az ajánló levél pedig a kapott jegytől ($P(A|J)$). Ezek ismeretében felírható a láncszabály a (2.1) képlet alapján:

$$P(N, I, J, P, A) = P(N)P(I)P(J|I, N)P(P|I)P(A|J)$$

A topológiai szemantikát a következő két egymással ekvivalens meghatározás bármelyike rögzíti (Russell & Norvig, 2005):

1. Egy csomópont feltételesen független a nem leszármazottaitól, feltéve, hogy a szülei adottak.
2. Egy csomópont feltételesen független az összes többi csomóponttól a hálózatban, a szülei, gyermekei és gyermeki szüleinek az ismeretében – azaz a Markov-takarójának ismeretében.

Ezeket a meghatározásokat mutatja be a 2.3. ábra. (Russell & Norvig, 2005)

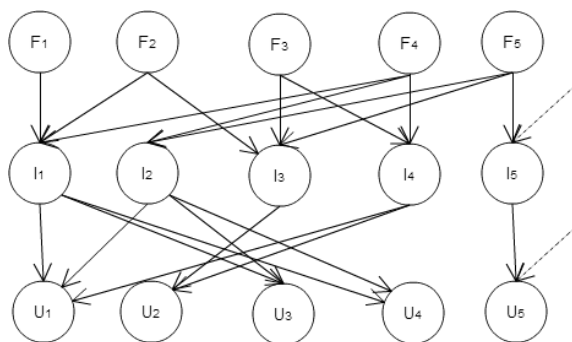


2.3. ábra Feltételes függetlenségi relációk BN-ben

2.3. ábra **a** része mutatja, hogy az X csomópont feltételesen független a nem leszármazottaitól (Z_{ij} –ktől) a szülei (a szürke területen látható U_i -k) ismeretében. A 2.3. ábra **b** része pedig azt az esetet mutatja be, mikor X csomópont feltételesen független a háló összes többi csomópontjától a Markov-takarójának ismeretében (a szürke terület).

2.2 Topológia, függőségi kapcsolatok MovieLens adathalmaz esetén

A rendelkezésre álló információk alapján első lépésben meg kell állapítani, hogy mik lesznek a Bayes háló csomópontjai. Mint korábban már említettem a BN-ben a csomópontok az állítások, az élek pedig a köztük lévő kapcsolatokat reprezentálják. Ezek alapján egyértelmű, hogy lesznek a műfajokhoz tartozó (F), a filmekhez tartozó (I) valamint felhasználók szavazatához tartozó csomópontok (U) a hálóban. Ezek közötti kapcsolatok is ilyen módon könnyen felírható. Ha egy filmet leír egy műfaj, akkor köztük kapcsolat lesz. Ha egy filmet pontoz egy felhasználó, akkor pedig közöttük lesz kapcsolat. Tehát egy $F \rightarrow I \rightarrow U$ összefüggés alakult ki. Ezek után felrajzolható a háló, a könnyebb elképzelés érdekében, melyet a következő 2.4. ábra reprezentál.



2.4. ábra Bayes háló

Minden egyes tulajdonsághoz létezik egy F csomópont, és minden egyes filmhez létezik egy I csomópont. Mindkettő bináris értékkel rendelkező csomópontok, $\{f_{k,0}, f_{k,1}\}, \{i_{j,0}, i_{j,1}\}$, ahol az alindex megmutatja, hogy az adott tulajdonság vagy film beleszámít-e a felhasználó preferenciájába – 1-es alindex, vagy sem – 0-s alindex. U_a felhasználók szavazatának valószínűségi eloszlását reprezentálja. U-ban $\#r + 1$ db állapot van (ahol $\#r$ az adható pontok száma), az általam vizsgált esetben egy 1-5-ig terjedő skála szerint pontoztak a felhasználók, és csak egész számokat adhattak, így $\{0, 1, 2, 3, 4, 5\}$ lesznek U állapotai.

Ezek a már ismert adatok. Ahogy korábban említettem a cél az, hogy minden film és minden felhasználó esetén meg legyen becsülve a pontszáma. Ehhez hibrid ajánlási technikát használva új csomópontok kerülnek a hálónkba: tartalom-alapú előrejelzés, kollaboratív előrejelzés, valamint hibrid előrejelzés. Ahány film van ugyanannyi tartalom alapú csomópont lesz, és ahány felhasználó annyi kollaboratív csomópont. A_{CB} –ként nevezem a tartalom alapú előrejelzés felhasználásának eredményét, A_{CF} -nek a felhasználó alapú kollaboratív szűrés által

számolt pontszám, és a A_H -nak pedig a hibrid eredményt. A_{CF} szülői azon felhasználók, akiknek hasonló az ízlése. A_{CB} szülei pedig azok a filmek, amelyek hasonlítanak egymáshoz. Két filmet akkor hasonló, ha legalább egy műfajban megegyeznek. A felhasználók közötti hasonlóság számítása két komponensből tevődik össze. Először Pearson-féle korrelációs számítással kiszámítható a páronkénti korreláció csak azon értékelési mintákra, ahol mindketten értékelték az adott filmet. Ehhez minden felhasználó átlag szavazatát ki kell számolni a (1.4) egyenlet segítségével, majd a Pearson-féle korrelációs együtthatót az (1.3) képlettel. A második komponens arra szolgál, hogy büntesse azokat a hasonlósági értékeket, ahol a magas korreláció annak köszönhető, hogy csak nagyon kevés közös elemet értékelték. Az így kapott pontosabb hasonlóságok számítására szolgáló képlet a (2.3) egyenlet, ahol az átlag szavazathoz Bayes esetén speciálisan a (2.2) egyenlet használható.

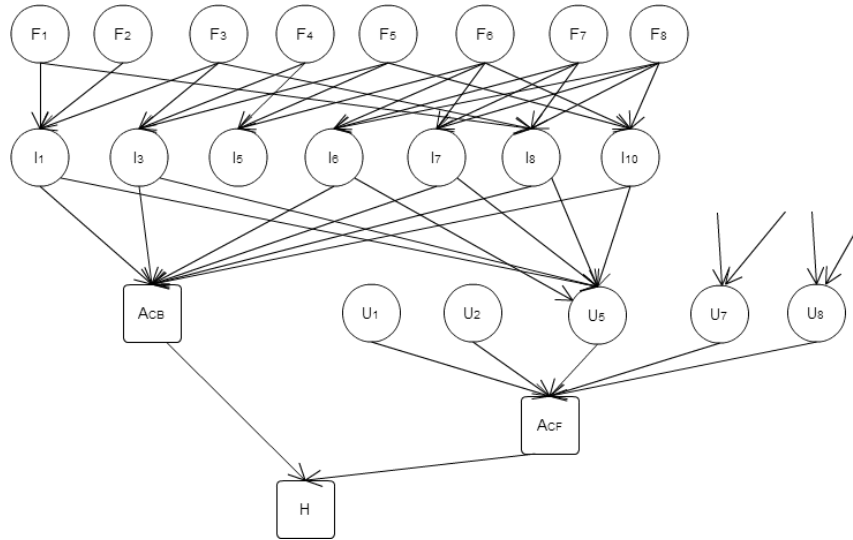
$$\bar{v}_i = \frac{1}{|Pa(U_i)|} \sum_{I_j \in Pa(U_i)} v_{ij} \quad (2.2)$$

$$Sim(u_i, u_k) = abs(Pearson(u_i, u_k)) * \frac{|I(u_i) \cap I(u_k)|}{|I(u_i)|} \quad (2.3)$$

Ahol a $|I(u_i) \cap I(u_k)|$ azoknak a filmeknek a száma melyet u_i és u_k is értékelt, $|I(u_i)|$ pedig amit u_i pontozott.

Az előrejelzésekhez az adott felhasználóhoz leginkább hasonló N db felhasználó szavazatait vizsgáltam kollaboratív szűrés esetén.

A 2.5. ábra reprezentálja az új csomópontokkal kibővített háló egy részét, ezáltal a függőségi kapcsolatok leolvashatóak.



2.5. ábra Példa a bővített BN-re

2.3 Súlyok számítása

Figyelembe véve a felhasználók számát, akik részt vesznek az előrejelzésben, kidolgozható egy kanonikus modell, amely reprezentálja a feltételes valószínűségi eloszlásokat. Ha feltételezzük ezt a modellt, akkor a feltételes valószínűségi táblázat kisebb részekre faktorizálható, és kombinálhatóak az értékeik.

Legyen X_i egy csomópont a BN-ben, melynek vannak szülői $Szülő_k(X_i)$, ahol a k . szülő Y_k . Ekkor a kanonikus súlyozott összeget felhasználva kiszámíthatjuk a feltételes valószínűségi eloszlását az adott X_i csomópontnak a (2.4) képlettel,

$$Pr(x_{i,j} | Szülő_k(X_i)) = \sum_{Y_k \in Pa(X_i)} w(y_{k,l}, x_{i,j}) \quad (2.4)$$

ahol $y_{k,l}$ Y_k értéke a $pa(X_i)$ konfiguráció esetén, $w(y_{k,l}, x_{i,j})$ pedig olyan súly, ami méri az Y_k változó l . értéke hogyan írja le az X_i csomópont j . állapotát. Ezek a súlyok nemnegatív számok, melyeket összeadva 1-et kapunk. Formálisan a (2.5) képlet írja le ezt a tulajdonságot.

$$\sum_{j=1}^r \sum_{Y_k \in Pa(X_i)} w(y_{k,l}, x_{i,j}) = 1 \quad (2.5)$$

Ezeket a súlyokat azért ajánlott használni, mivel ezáltal a becslés pontosabb lesz mint az általános modellek esetén. Minden olyan szinten számolnunk kell súlyokat, melyeknek van őse.

Minden F –hez rögtön megadható a hozzá tartozó valószínűségi eloszlás, mivel nincs ősök, tehát nincs szükségünk kanonikus súly számítására. Az F -ek úgynevezett prior valószínűségek. Mivel minden F csomópont két állapot (f számít – f nem számít), ezért mindkettőnek meg kell adni a valószínűségét. A két valószínűség összege 1, ezért elég csak az egyiket kiszámítani, majd abból könnyedén egy kivonással számítható a másik (2.6) képlet alapján.

$$\Pr(f_{k,0}) = 1 - \Pr(f_{k,1}) \quad (2.6)$$

A valószínűség kiszámítására két alternatíva áll rendelkezésre, az első, amikor minden műfajt azonosan fontos (EQ), ekkor a (2.7) képlet használható,

$$\Pr(f_{k,1}) = \frac{1}{nf} \quad (2.7)$$

ahol nf a műfajok száma. Ha a műfajok különböző súllyal rendelkeznek (RF), akkor pedig a (2.8) képlet használható,

$$\Pr(f_{k,1}) = \frac{nk + 0.5}{nm + 1} \quad (2.8)$$

ahol nk az a szám, hogy hány filmet ír le az adott műfaj, nm pedig a filmek száma.

Mivel az I csomópontok is binárisak, így itt is két valószínűségi érték fog tartozni minden I -hez. Az I -nek van őse, ezért ebben az esetben először súlyokat kell számolni. A műfaj alindexe alapján a súlyok (2.9) egyenlet alapján számítható,

$$w(f_{k,1}, i_{j,1}) = \frac{\log\left(\frac{nm}{nk} + 1\right)}{M(I_j)} \quad (2.9)$$

$$w(f_{k,0}, i_{j,1}) = 0$$

ahol $M(I_j)$ normalizáló faktor, melyet a (2.10) képlettel számítjuk:

$$M(I_j) = \sum_{F_k \in Pa(I_j)} \log((n_{jk}^{nm}/n_k) + 1) \quad (2.10)$$

Ezután a (2.4) képletet felhasználva kiszámítható az adott csomópontához tartozó feltételes valószínűségek $Pr(i_{j,1}|pa(I_j))$ és (2.6) képlethez hasonlóan $Pr(i_{j,0}|pa(I_j))$ -et.

A következő szinten azt kell vizsgálni, hogy melyik film mennyire befolyásolja az adott felhasználó szavazatát. Ehhez szintén súlyokat kell számolni. Ez két részre bomlik, az első a tartalom alapú predikcióhoz tartozó súlyok, másik pedig a kollaboratív szűréshez tartozó becslés.

Tartalom alapú becslés esetén meg kell vizsgálni az elemek hatását az értékelési mintára. Figyelembe véve hogy a filmfontos vagy sem a felhasználó számára (alindexe) a súlyok a (2.11) alapján számíthatóak:

$$\begin{aligned} w(i_{k,1}, u_{CB,s}) &= 1/|Pa(U_a)| \\ w(i_{k,1}, u_{CB,t}) &= 0 \end{aligned} \quad (2.11)$$

$$\begin{aligned} w(i_{k,0}, u_{CB,0}) &= 1/|Pa(U_a)| \\ w(i_{k,0}, u_{CB,t}) &= 0 \end{aligned}$$

Ahol 't' és 's' pontszámok úgy, hogy 's' az amit az U_a felhasználó adott pontszám $1 \leq s \leq \#r$, $t \neq s$, $0 \leq t \leq \#r$, és $\#r$ az adható pontok száma. Ezután számíthatóak a $Pr(u_{a,k}|pa(u_a))$ értékek szintén a (2.4) alapján.

A kollaboratív alapú előrejelzés esetén meg kell határozni, hogy mely hasonló felhasználó mekkora súllyal járul hozzá a célfelhasználó pontszámának előrejelzéséhez. Minden hasonló felhasználóra számítani kell a súlyokat. A súlyok több részből tevődnek össze, kiszámítását a (2.12) képlet mutatja be,

$$w(u_{i,t}, a_{CF,s}) = \text{RSim}(U_i, A) * \text{Pr}^*(A = s | U_i = t)$$

$$w(u_{i,t}, a_{CF,0}) = 0$$

(2.12)

$$w(u_{i,0}, a_{CF,0}) = \text{RSim}(U_i, A)$$

$$w(u_{i,0}, a_{CF,s}) = 0$$

ahol $1 \leq t, s \leq \#r$, $\text{RSim}(U_i, A)$ a relatív minőség (hasonlóság) amely minden felhasználó és szülei között számítandó a (2.13) képlettel, $\text{Pr}^*(A = s | U_i = t)$ pedig annak a valószínűsége, hogy a felhasználó s -re értékelt egy filmet, amíg az őse t -re. Utóbbi számításához a (2.14) képlet áll rendelkezésre.

$$\text{RSim}(U_i, A) = \frac{\text{Sim}(U_i, A)}{\sum_{j \in \text{Pa}(A_{CF})} \text{Sim}(U_j, A)}$$

(2.13)

$$\text{Pr}^*(A = s | U_i = t) = \frac{N(u_{i,t}, a_s) + \frac{1}{\#r}}{N(u_{i,t}) + 1}$$

(2.14)

Ahol $N(u_{i,t}, a_s)$ azoknak a filmeknek a száma, amelyek az $I(u_i) \cap I(A)$ -ból $U_{i,t}$ -re, A pedig s -re értékelt. $N(u_{i,t})$ pedig azoknak a filmek száma az $I(u_i) \cap I(A)$ -ból, amelyet U_i értékelt t -re.

A topológia legalsó szintjén helyezkedik el a hibrid ajánlati érték, melynek két szülője is van, a kollaboratív és a tartalom alapú szűrés, $\text{Pr}(A_H | A_{CF}, A_{CB})$. Bevezethető egy alfa paramétert, amely $0 \leq \alpha \leq 1$, ezzel lehet ellenőrizni mindkettő komponens hozzájárulását a hibrid pontszámhoz. Ekkor (2.15) egyenlet használható fel,

$$\text{Pr}(a_{h,s} | a_{CB,s}, a_{CF,s}) = 1$$

$$\text{Pr}(a_{h,s} | a_{CB,s}, a_{CF,q}) = \alpha_j$$

(2.15)

$$\text{Pr}(a_{h,s} | a_{CB,t}, a_{CF,s}) = 1 - \alpha_j$$

$$\Pr(a_{h,s} | a_{CB,t}, a_{CF,q}) = 0$$

ahol $0 \leq s, q, t \leq \#r$ és $s \neq q \neq t$.

Minél magasabb az α értéke, annál nagyobb a tartalom alapú komponens súlya. Ha az α értéke 0, akkor a hibrid modell tisztán kollaboratív modell lenne. Éppígy ha $\alpha=1$, akkor pedig tisztán tartalom alapú modell lenne. A közbülső értékek esetén mindkét információs figyelembe kell venni különböző súlyokkal. Az α értéke ajánlórendszerenként eltérő lehet, úgy kell megválasztani, hogy maximalizálja az ajánlás minőségét.

Felhasználva

- 1) hogy a BN-ben a csomópontok függetlenek az összes őstől, ha tudjuk a szülők értékét és
- 2) A kanonikus modell előnyeit,

a feltételes valószínűségi eloszlás kifejezésével ki lehet számolni a végső valószínűséget top-down mechanizmussal. Az előbbieken felvázolt topológiában három csomópont szint van (tulajdonságok, filmek, felhasználók), és ez kiegészítve még két becsléssel a felhasználók esetén. A posterior valószínűségek kiszámításának tétele segít kiszámítani a valószínűségeket. Ezáltal minden csomópont tartalmazni fogja a feltételeket az őseiktől.

Legyen X_a egy csomópont az BN-ben, m_{X_a} az X_a szüleinek a száma, Y_j egy csomópont a $\text{Pa}(X_a)$ -ból, l_{Y_j} pedig az Y_j által felvett állapotok száma. Ha a feltételes valószínűség kifejezhető a feltételek mellett a (2.4) képlet segítségével akkor a pontos posterior valószínűség kiszámítható a (2.16) képlettel.

$$\Pr(x_{a,s} | ev) = \sum_{j=1}^{m_{X_s}} \sum_{k=1}^{l_{Y_j}} w(y_{j,k}, x_{a,s}) * \Pr(y_{j,k} | ev) \quad (2.16)$$

Tartalom alapú komponens esetén fontos, hogy a feltétel a tulajdonságok egy halmaza, vagy pedig egy elem. Előbbi esetben az első tétel alkalmazható, (2.16) képlet, második esetben pedig ki kell számítani a $\Pr(f_{k,1} | i_{j,1})$ valószínűségeket minden F_k csomópont esetén amely kapcsolódik a cél elemmel. A következő tétel mondja el, hogy hogyan kell ebben az esetben eljárni.

Legyen F_k az I_j csomópont szülője a BN-ben, ami egy gyökér csomópont a BN-ben. A releváns műfaj posterior valószínűsége adott I_j változó esetén a (2.17) alapján számítható ki,

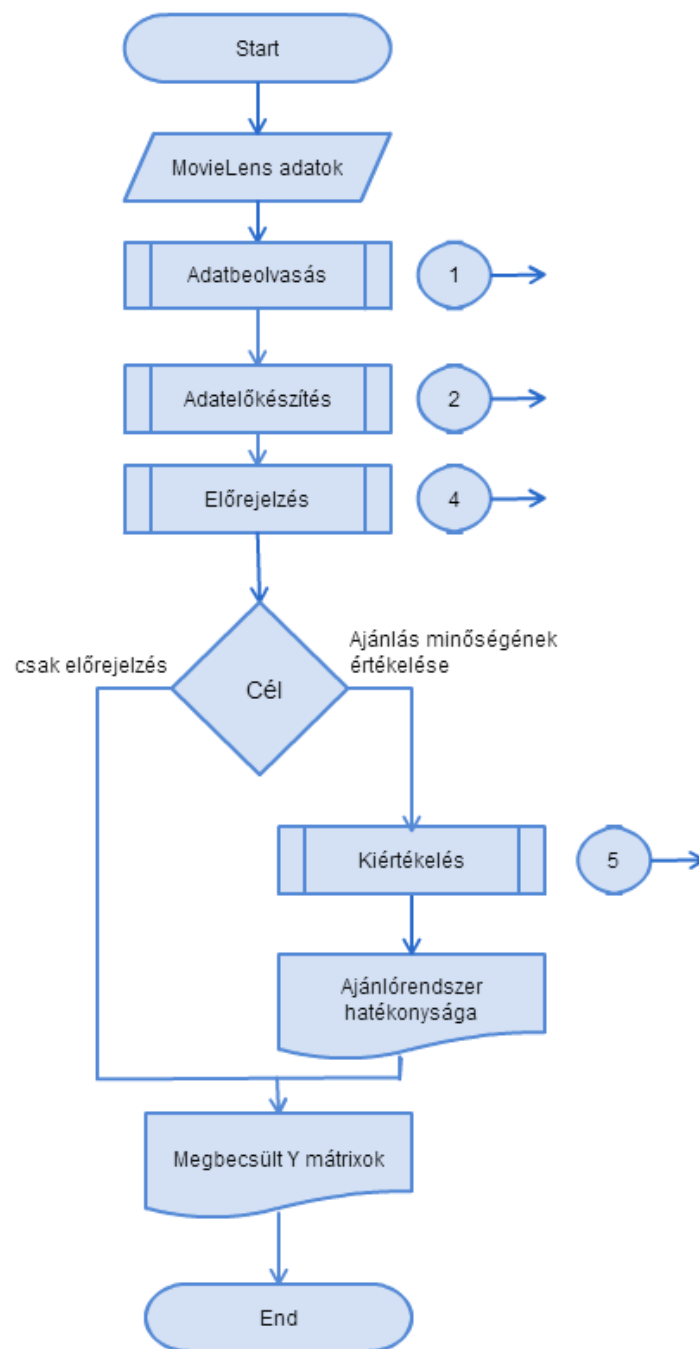
$$\Pr(f_{k,1}|i_{j,1}) = \begin{cases} \Pr(f_{k,1}) & \text{ha } F_k \notin Pa(I_j) \\ \Pr(f_{k,1}) + \frac{w(f_{k,1}, i_{j,1}) \Pr(f_{k,1}) (\Pr(f_{k,0}))}{\Pr(i_{j,1})} & \text{ha } F_k \in Pa(I_j) \end{cases} \quad (2.17)$$

ahol $F_k \in Pa(I_j)$ esetén a számlálót a (2.18) képlet adja meg.

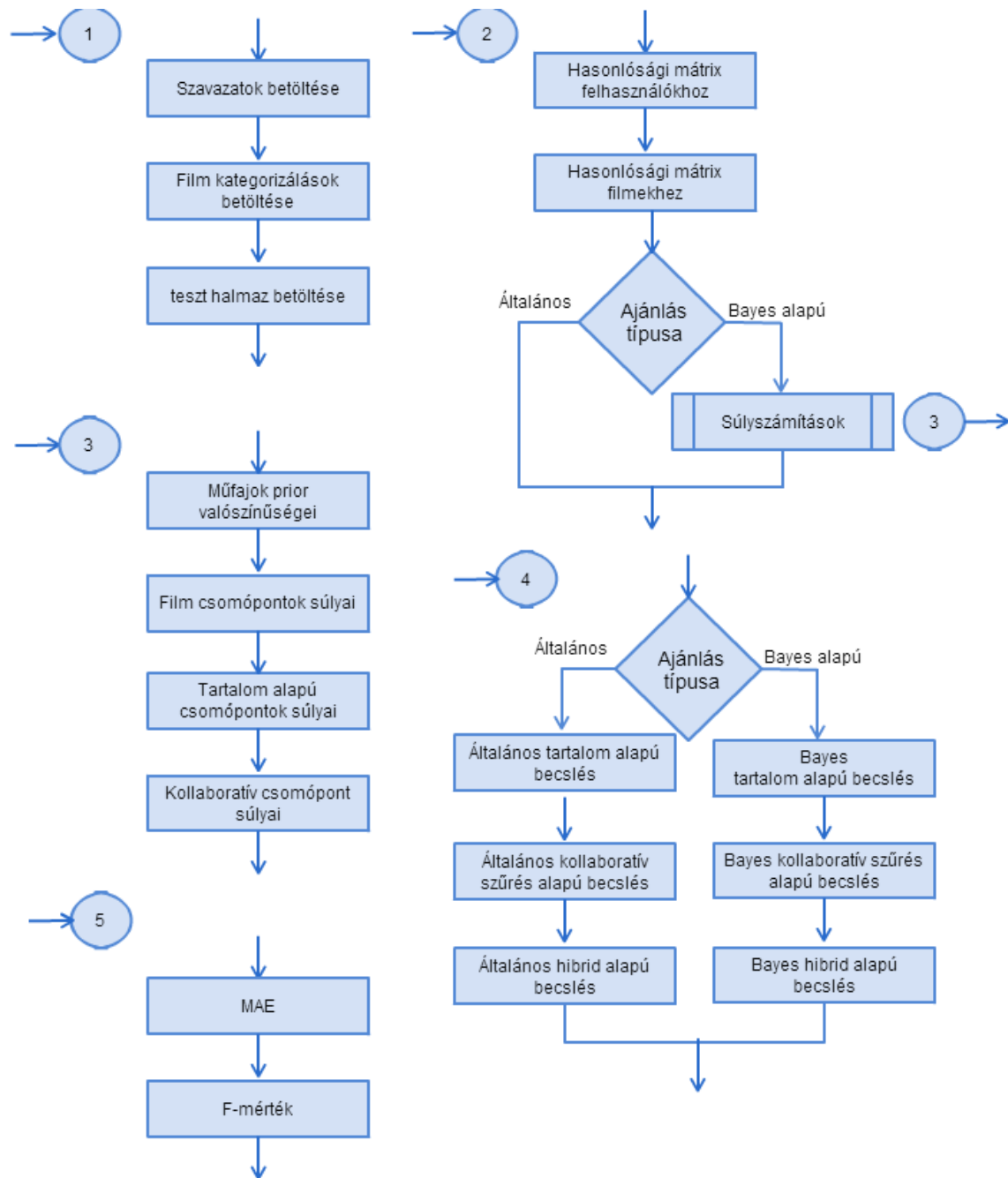
$$\Pr(i_{j,1}) = \sum_{F_k \in Pa(I_j)} w(f_{k,1}, i_{j,1}) * \Pr(f_{k,1}) \quad (2.18)$$

2.4 Implementálás MATLAB-ban

A MovieLens 100k adathalmazában lévő adatok közül fontos, hogy hányadik felhasználó hányadik filmet hányasra értékelte, illetve hogy hányadik filmet hányadik tulajdonságok írják le. Összesen 943 felhasználó (nu) és 1682 film (nm) szerepel ebben az adatbázisban, a filmeket 18 műfajba (nf) sorolják be. Van olyan film is, melyről még nincs megadva, hogy melyik kategóriába is tartozik, jelenleg „ismeretlen” műfajba lett sorolva. A saját ajánlórendszerem implementálásának menetét a 2.6. ábra és 2.7. ábra szemlélteti.



2.6. ábra Implementálás Folyamatábrája 1.



2.7. ábra Implementálás Folyamatábrája 2.

Az adatok betöltéséhez három függvényt készítettem, az első beolvasta, hogy hanyadik filmet hanyadik felhasználó értékelte már, és mi volt az értékelés eredménye. Ezáltal felépítettem egy $\mathbf{Y}$ mátrixot, ami egy $nm * nu$ méretű mátrix, $y_{ij} = s$, ahol s az adott pontszám illetve 0, tehát $0 \leq s \leq 5$, ahol az $y_{ij} = 0$ azt jelenti hogy az i . filmet a j . felhasználó még nem értékelte. Segédmátrixként létrehoztam még egy $\mathbf{R}$ bináris mátrixot, mely szintén $nm * nu$ méretű, ahol $r_{ij} = 1$ ha az i . filmet a j . felhasználó pontozta, és $r_{ij} = 0$ ha nem. A másik függvényem a műfajbesoroláshoz kötődik. Egy $nm * nf$ méretű $\mathbf{D}$ mátrixot hoztam létre,

amely szintén bináris. $d_{ik} = 1$, ha az i . filmet leírja a k . műfaj, $d_{ik} = 0$, ha nem. A harmadik függvényem betöltötte a teszteléshez szükséges validálási halmazt, amely az eredeti adathalmaz rekordjainak 20%-át tartalmazza. A mátrixot **Y_test**-nek neveztam el.

Az adatok betöltése után a következő függvényem a felhasználók közötti hasonlóságokat számolja. A hasonlóságokat egy $nu * nu$ méretű **H** mátrix tárolja, értékei $[0,1]$ közötti intervallumba esnek. Minél magasabb az érték, annál magasabb a hasonlóság. A hasonlósági mértékek kiszámításához a (2.3) képletet használtam fel. A filmek közötti hasonlóságot az $nm * nm$ méretű **S** bináris mátrixban tároltam. Filmek közötti hasonlósághoz azt a teóriát alkalmaztam, miszerint azok a filmek hasonlóak amelyek legalább egy műfajukban megegyeznek.

Implementáltam az általános ajánlási technikákat is a Bayes alapú mellett. A Bayes alapú rendszerekhez szükséges súlyokat számító függvényem megadja a műfajokhoz tartozó prior valószínűségeket (2.6), (2.7) és (2.8) képleteket felhasználva, mely **F** néven RF esetén, **F2** néven EQ esetén. Az össel rendelkező csomópontok esetén kiszámolta a hozzájuk tartozó súlyokat: filmek esetén (2.9) képlettel **Wi**, felhasználói szavazatok esetén két féle súlyt számoltam, az első a **Wu**, melyet a (2.11) képlettel számoltam, a másik pedig a kollaboratív csomópont **Wv**, amihez a (2.12) képletre volt szükségem.

Bayes háló alapú algoritmusok esetén a tartalom alapú előrejelzéskor a $Pr(u_{cel,s}|I_k)$ valószínűségeket kerestem minden s -re, ahol k egy adott filmindex, s pedig az értékelési pont. A valószínűségi eloszlások meghatározásához a (2.16) képletet használtam minden I_k -hoz hasonló filmek esetén. A $Pr(I_{hasonlo_j})$ meghatározása után (ahol $j = 0, \dots, \text{hasonlo filmek száma}$) már számolhattam szintén a (2.16) képlet segítségével a kérdéses valószínűségeket. Tartalom alapú előrejelzést EQ (minden műfaj azonos valószínűséggel bír) és RF (műfajok súlyozottak) esetén is számoltam. Kollaboratív szűrés esetén a $Pr(v_{a,s}|I_k)$ értéket keressük. Ehhez először a (2.16) képletet felhasználva kiszámítottam N darab hasonló felhasználó esetén a $Pr(u_{hasonlo_j,s}|I_k)$ valószínűségeket minden s esetén (ahol $j = 0, \dots, N$). Ezután szintén a (2.16) képlettel számítottam ki a keresett valószínűségek eloszlását. A hibrid technika esetén a korábbi két technikával kapott pontszámokhoz tartozó valószínűségi eloszlásait súlyoztam egy alfa paraméterrel, melynek értéke $[0,1]$ intervallumból kerül ki. Így megkaptam a hibrid ajánlás esetén az értékelési pontokhoz tartozó valószínűségi eloszlásokat. Legvégül a valószínűségek alapján megad

határoztam az előre jelzett pontszámot, mind mediánszámítással, mind pedig a legvalószínűbb értéket kiválasztásával.

Általános algoritmusoknál nincs szükségünk a korábban számított súlyokra. Ekkor tartalom alapú ajánlaskor a célfelhasználó esetén megvizsgáltam a célfilmmel hasonló filmekre adott pontszámát, és azok alapján becsültem egy pontszámot. Kollaboratív szűrés esetén pedig a célfelhasználóhoz hasonló felhasználók pontszámát vizsgáltam meg a célfilm esetén, és azok alapján becsültem meg az értékelést.

3. Kísérlet, alkalmazás

Ebben a fejezetben olvasható a kísérletem eredményei. Célkitűzésem több komponensből tevődött össze. Az általam tanulmányozott irodalmak nem minden általam érdekesnek gondolt eredményt emeltek ki, ezáltal bővíteni kívántam az elemzéseket, hozzájuk grafikus megjelenítést is társítva. Igazolni kívántam az implementáció megfelelőségét és hibamentességét. Tanulmányoztam a számítási idő változását szomszédok számának változásával kollaboratív szűrés esetén. Összehasonlítottam az általános és a Bayes alapú technikákat két féle módszer segítségével. Kollaboratív ajánlaskor figyeltem az ajánlás minőségének változását különböző számú legközelebbi szomszéd esetén. Tartalom alapú eljárásnál pedig az RF és az EQ esetekben hasonlítottam össze. Megvizsgáltam a tartalom alapú ajánlások korrelációja a kollaboratív ajánlásokkal. Megfigyeltem hibrid ajánlás esetén az alfa változtatásával a predikció pontosságának alakulását. Hibrid módszernél az előzőekben ismertetett szempontokon kívül még az optimális alfa paramétert is kerestem, amely a kollaboratív és a tartalom alapú szűrés súlyát határozta meg.

Vizsgálataimat egy elismert, széles körben alkalmazott benchmark adatbázis felhasználásával tettem meg amely lehetővé teszi a széles körű összehasonlítást is. Az alfejezetekben először magáról az adathalmazról lesz szó, majd pedig az elvégzett kísérleteket részletezem, illetve az eredményeimet ismertetem (GroupLens, 2003).

3.1 Probléma ismertetése

A MovieLens (ML) adathalmazát a Minnesotai Egyetem gyűjtötte össze GroupLens Research Project néven 1997. szeptember 19. és 1998. április 22. között. A filmekre egy 1-5 terjedő skálán lehetett szavazni (1=szörnyű, 2=eléggye rossz, 3=jó, 4=élvezetes, 5=ezt muszáj megnézni) filmművészeti kontextusban. Az összegyűjtött adatok ingyenesen elérhető a

GroupLens honlapjáról. A letöltött anyagból felhasználtam az u.item nevű fájlt, amely tartalmazza, hogy melyik film mely kategóriákba tartozik; az u.base-t, melyben az adott felhasználók adott filmre adott pontszámai vannak. Ezeken felül az u.test fájl tartalmaz egy 20%-os teszt halmazt, ami az összehasonlítások miatt volt hasznos számomra.

A közzétett adathalmazok közül az ML 100k-t választottam ahhoz, hogy elvégezhessem az algoritmusok tesztelését. Ebben az adathalmazban 943 felhasználó és 1682 film szerepelt, összesen 100 000 értékelést tartalmazva (1-5 terjedő skálán). A filmeket 18 műfajba sorolták be (akció, kaland, animációs, gyerekfilm, vígjáték, krimi, dokumentumfilm, dráma, fantasztikus, film-noir, horror, musical, misztikus, romantikus, sci-fi, thriller, háborús és western). Volt olyan film is, melyről még nem volt megadva hogy melyik kategóriába is tartozik, jelenleg „ismeretlen” műfajba lett sorolva. Az ehhez tartozó validálási halmaz az eredeti adathalmaz rekordjainak 20%-át tartalmazza, tehát 20 000 értékelés szerepel benne.

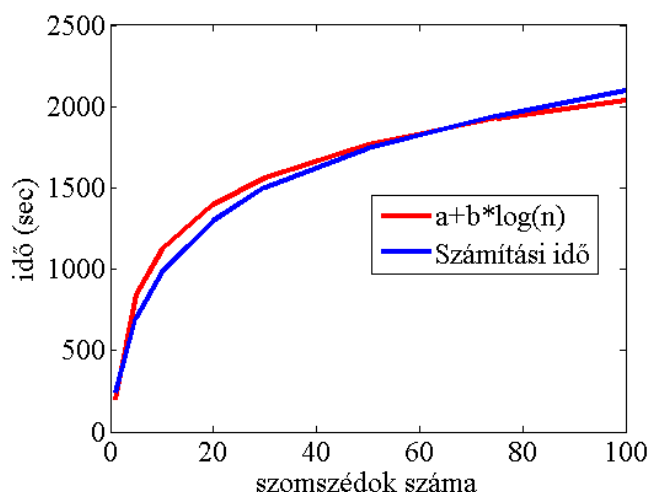
3.2 Az algoritmus skálázhatóságának vizsgálata

Elsőként a kollaboratív szűrés számítási idejének változását vizsgáltam a szomszédok számának (NS) növekedésével. A számítási idő részletes számítása helyett másodpercben mérve a MATLAB beépített stopperórája segítségével nyert eredményeket mutatom be. A kísérletet Windows 7 rendszer 32 bites verzióján végeztem R2009b MATLAB-ot használva. A számítógép Pentium(R) Dual-Core T4500-as 2*2.30 GHz-es processzorral, 3.00 GB RAM-mal rendelkezik. Az eredményeket a 3.1. táblázat tartalmazza.

3.1. táblázat Számítási idő CF esetén

NS	Bayes	Általános
1	242.445721	12.430224
5	699.830669	21.089786
10	979.151548	25.223792
20	1295.163571	31.480355
30	1503.352174	35.076507
50	1744.461285	38.838852
75	1944.312915	41.213898
100	2099.605353	43.906075

BN alapú ajánlás esetén a számítási idő egy nagyságrenddel több, mivel a súlyok felhasználása, kombinálása több időt vesz igénybe, mint egy egyszerű átlagolás. Ajánlórendszereknél az ajánlás, előrejelzés nem időkritikus, ezért a Bayes alapú technika kivitelezhető. A számítási időt grafikonon ábrázolva még jobban látható a változás. Mivel vizsgálatom középpontjában a Bayes módszer áll, ezért az ahhoz tartozó függvényt mutatom be a 3.1. ábra segítségével. Jól látható, hogy a szomszédok számával a számítási idő nem exponenciálisan növekszik, inkább $\log n$ -re utal.



3.1. ábra Számítási idő BN alapú CF esetén

3.3 Általános és Bayes alapú technikák összehasonlítása

Ebben az alfejezetben az általános és a Bayes alapú technikák összehasonlításának eredményét ismertetem. Bayes alapú modell esetén tartalom alapú ajánláskor az RF és EQ eseteket hasonlítottam össze egymással és az általános eljárással. Kollaboratív szűrés esetén az alap és Bayes módszernél is a szomszédok számának változtatásával készítettem az összehasonlításokat. Mindegyik esetben a MAE értékeket és az F mérték alakulását vizsgáltam.

Elsőként a MAE-t használtam fel az összehasonlításra. A tartalom alapú ajánlások hatékonyságát mérésének az eredményét a 3.2. táblázat mutatja. Ekkor a pontszám megállapításakor Bayes esetben a legvalószínűbb pontszámot (1. oszlop) és a mediánértékhez tartozó pontszámot (2. oszlop) is választottam becsült értéknek. Általános ajánláskor pedig a célfelhasználó által a hasonló filmekre adott értékelések átlagát választottam ki becsült értéként.

3.2. táblázat Tartalom alapú ajánlás MAE értékei

	Bayes – legval.	Bayes - medián	Általános
EQ	0.8556	0.7940	0.8023
RF	0.8638	0.7998	

Látható, hogy ha a műfajok nem súlyoztam (EQ), akkor jobb eredményt kaptam, mint ha minden műfajt más-más valószínűségi értékkel vettem volna figyelembe (RF). Ha a mediánértékhez tartozó pontszámot becsültem, akkor jobbnak bizonyult a Bayes alapú technika, mint az általános módszer.

Ezt követően a kollaboratív ajánlásokat vizsgáltam szintén MAE értékek felhasználásával. Ezt különböző számú legközelebbi szomszédal végeztem el, 1-5-10-20-30-50-75-100 legközelebbi szomszéd bevonásával figyeltem a predikció pontosságának a változását mind az alap, mind pedig a Bayes ajánlórendszerem esetén. Eredményeimet a 3.3. táblázat mutatja.

3.3. táblázat Kollaboratív szűrés MAE értékei

NS	Bayes-legval.	Bayes-medián	Általános
1	1.2046	0.9983	0.8522
5	0.7040	0.7046	0.7627
10	0.6301	0.6492	0.7347
20	0.5932	0.6120	0.7223
30	0.5768	0.6009	0.7194
50	0.5591	0.5876	0.7228
75	0.5503	0.5816	0.7288
100	0.5419	0.5760	0.7339

Az általános módszer esetén az látható, hogy a szomszédok számának növekedése egy ideig javítja a predikciót, de 30 és 50 szomszédság közötti valamely szomszédszámtól már zajként tekinthető a nagyszámú szomszédság. Bayes alapú technikák esetén jobb eredményt kaptam akár a legvalószínűbb, akár a mediánértékhez tartozó pontszámot választottam. A szomszédok számának növekedése a becslésem pontosságát javította, 100 szomszéd bevonásáig nem jelentkezett zaj.

A MAE vizsgálat után elvégeztem az F-mérték számítását. Ezen vizsgálatokat szinten külön megvizsgáltam kollaboratív szűrés és tartalom alapú ajánlás esetén is. Hasonlóan az előző kiértékeléshez most is tartalom alapú szűrés esetén EQ és RF eseteket is vizsgáltam, kollaboratív szűrésnél pedig különböző szomszédszámok esetét vizsgáltam. A top-N halmazba az átlagosan legjobb pontszámot elért felső 5% filmet raktam, ami 85 darab film volt. Eredményeimet a 3.4. táblázat és 3.5. táblázat tartalmazza.

3.4. táblázat F mérték CB esetén

	Bayes – legval.	Bayes - medián	Általános
EQ	0.0706	0.0824	0.1176
RF	0.1059	0.0824	

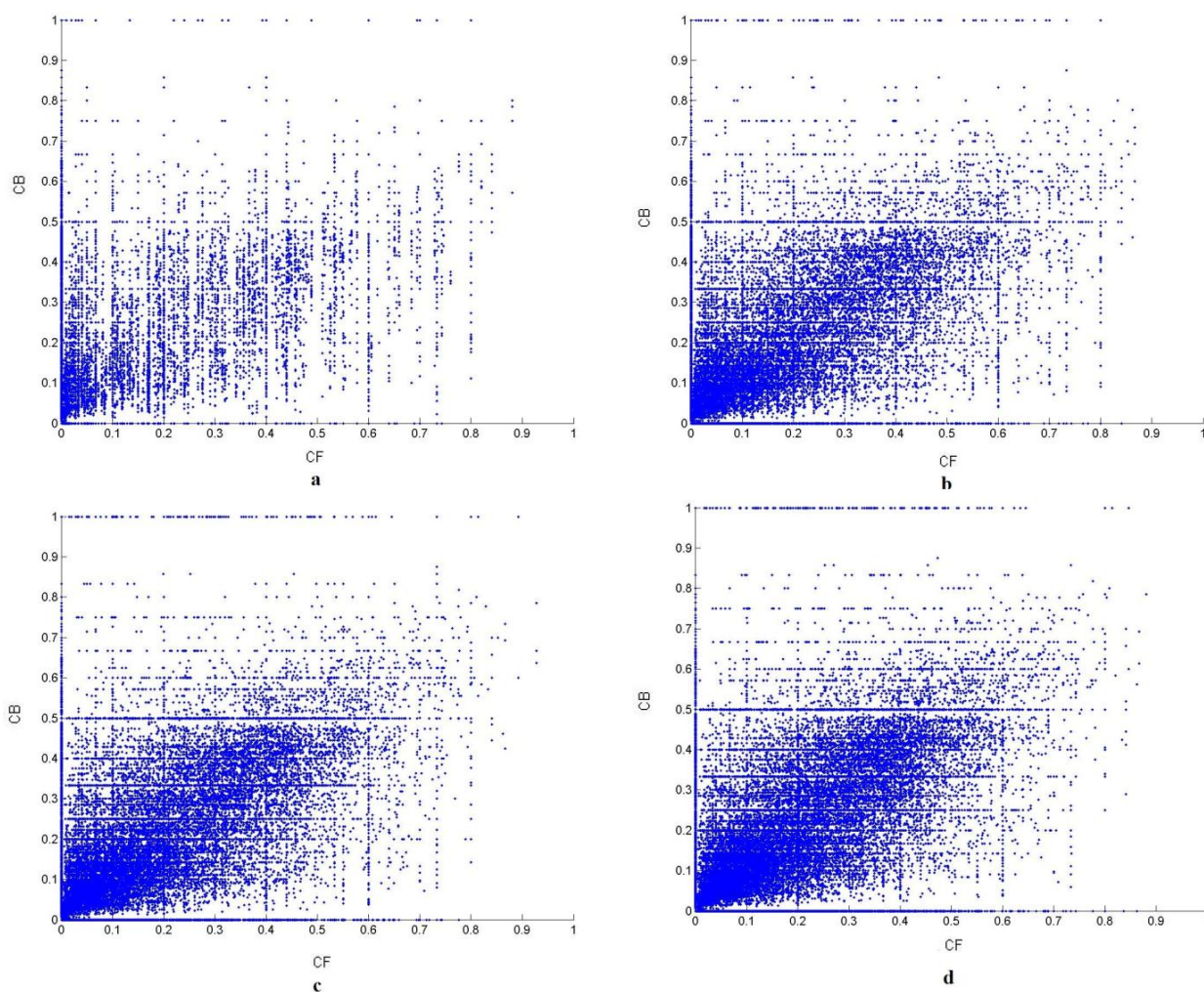
3.5. táblázat F mérték CF esetén

NS	Bayes-legval.	Bayes-medián	Általános
1	0.2706	0.1059	0.5294
5	0.3294	0.1059	0.6353
10	0.4588	0.1059	0.6706
20	0.5412	0.1059	0.6941
30	0.5765	0.1059	0.6824
50	0.6118	0.1059	0.6941
75	0.6353	0.0941	0.7176
100	0.6353	0.0941	0.6941

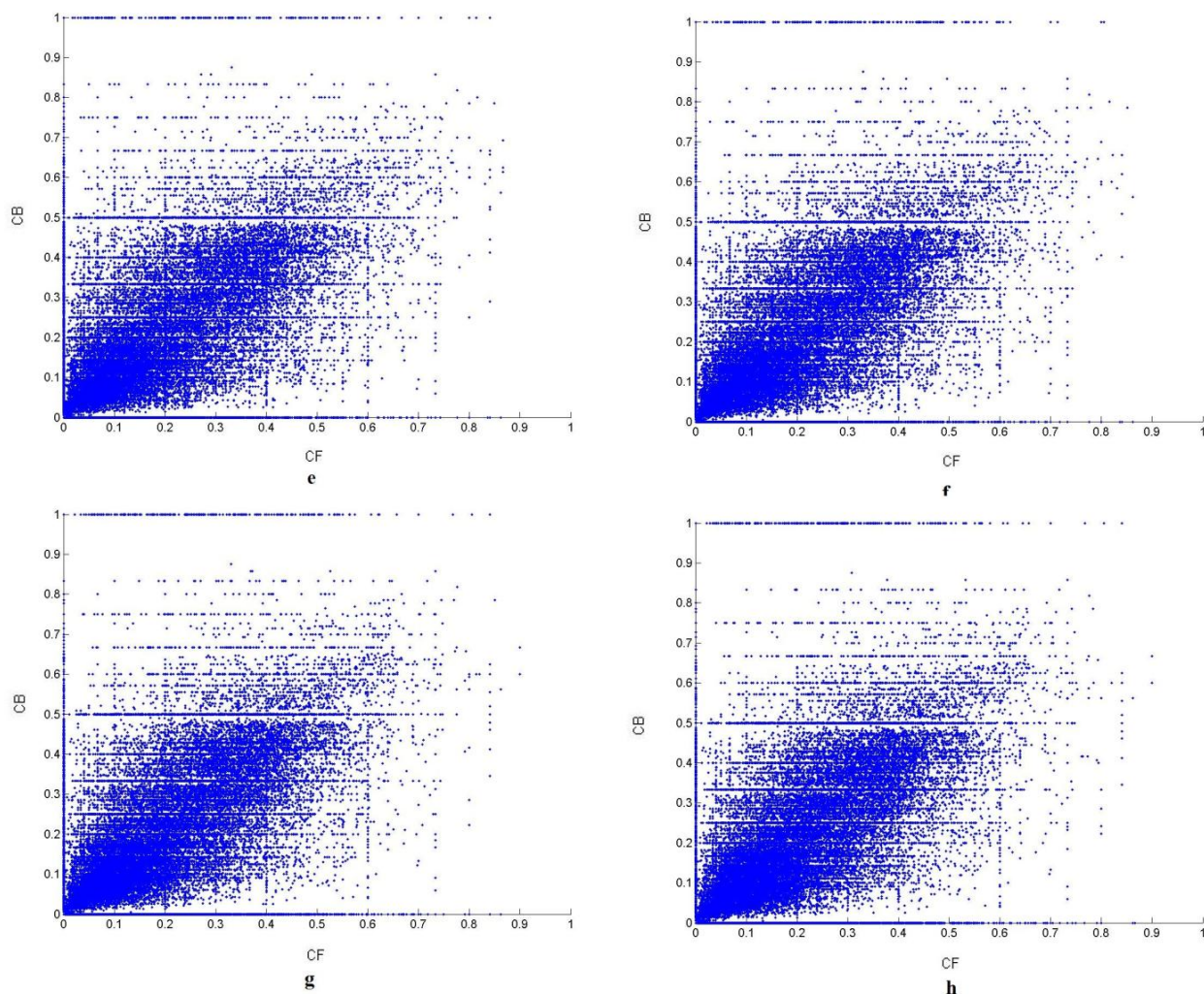
Az általános módszer azért bizonyult jobbnak, mivel általános módszerek esetén alapból átlagértékeket fogadtam el előre jelzett értékeknek. Kiértékeléseim eredményéből látható, hogy ha mediánértékhez tartozó pontszámot választom becsült értéknek, akkor rosszabb ajánlást kapok CF esetén, mint ha a legvalószínűbb pontszámot választottam volna. CB esetén átlagban minimális a legvalószínűbb módszer szintén jobbnak bizonyult. Mivel olyan ajánlórendszer fejlesztése a célom, amely nem csak pontos előrejelzést készítsen, hanem pontos ajánlásra is képes legyenek, ezért a későbbiekben már ennek a vizsgálatnak az eredményeként a legvalószínűbb pontszám kiválasztását alkalmaztam. A legvalószínűbb pontszám kiválasztása jobban tükrözi a Bayes alapelvét is.

3.4 Tartalom alapú és kollaboratív ajánlás korrelációja

Ebben az alfejezetben a Bayes technikán alapuló két módszer, a tartalom alapú és a kollaboratív előrejelzéssel nyert pontszámokhoz tartozó valószínűségi értékek korrelációját vizsgáltam a szomszédok számának változásával. Mivel az EQ és RF esetek közel azonos tendenciát mutatnak, ezért tartalom alapú szűrést csak EQ esethez tartozó eredményeimet szemléltetem, mivel abban az esetben kissé pontosabb volt a predikció, mint ha a műfajokat különböző súllyal súlyoztam volna. A korreláció szemléltetéséhez pontdiagramot használtam, ahol véletlenszerűen kiválasztott 125 filmhez és 70 felhasználóhoz tartozó valószínűségi eloszlásokat vizsgáltam az öt pontszám esetén. A 3.2. ábra és a 3.3. ábra mutatja be a korreláció változását a szomszédok számának növekedésének függvényében. Az **a**-nál 1, **b**-nél 5, **c**-nél 10, **d**-nél 20, **e**-nél 30, **f**-nél 50, **g**-nél 75, végül **h**-nél 100 legközelebbi szomszéd esete látható.



3.2. ábra CF - CB korreláció 1.



3.3. ábra CF - CB korreláció 2.

A fenti két ábrán látható, hogy a szomszédok számának növekedésével a becslés egyre konzisztensebbé vált. A szomszédság növelésével a két technika közötti korreláció is növekedett. A két technika nem teljesen korrelál egymással. Ez a felismerés is megerősíti annak a lehetőségét, hogy e két technikát ötvözzem. A kollaboratív szűrés és a tartalom alapú szűrés fúziójának eredményeit a következő 3.5. alfejezetben mutatom be.

3.5 Hibridizáció

Ebben az alfejezetben a korábban ismertetett két Bayes alapú algoritmus hibridizációjának elvégzésekor kapott eredményeket ismertetem. A hibrid technika esetén a kollaboratív szűréssel és a tartalom alapú ajánlással kapott pontszámokhoz tartozó valószínűségi eloszlásait súlyoztam egy alfa paraméterrel. Az alfa paraméter jelentéséhez a (2.15) képletet használtam fel, miszerint ha $\alpha = 1$ akkor tisztán tartalom alapú ajánlás történik, míg ha $\alpha = 0$, akkor tisztán kollaboratív szűrés valósul meg. Az így kapott hibrid ajánlórendszer esetén az

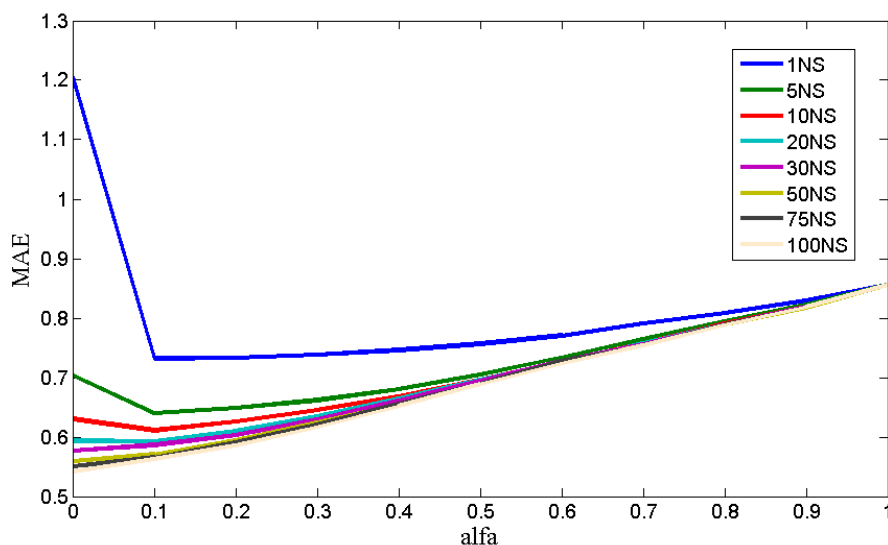
értékelési pont előrejelzésekor a legvalószínűbb pontszámot választottam. Az előrejelzés pontosságának (MAE) változását a szomszédok számának (NS) és az alfa paraméter értékének függvényében vizsgáltam, ábrázoltam. Korábban kapott eredmény miatt, miszerint ha minden műfaj esetén azonos valószínűséget feltételezek, akkor pontosabb becslést kapok, a hibrid modellem tartalom alapú komponense esetén az EQ technikát választottam. A 3.6. táblázatban foglaltam össze a MAE értékek változását az alfa és a szomszédok számának változásának függvényében, majd az eredményeket a 3.4. ábra grafikusán is megjelenítettem, értékeltem.

3.6. táblázat Hibrid modell MAE értékei

α Ns	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
1	1.2046	0.7322	0.7333	0.7382	0.7457	0.7563	0.7703	0.7904	0.8085	0.8302	0.8556
5	0.7040	0.6401	0.6484	0.6615	0.6806	0.7045	0.7339	0.7648	0.7940	0.8221	0.8556
10	0.6301	0.6110	0.6258	0.6451	0.6688	0.6962	0.7264	0.7585	0.7904	0.8194	0.8556
20	0.5932	0.5923	0.6105	0.6345	0.6644	0.6958	0.7240	0.7571	0.7888	0.8187	0.8556
30	0.5768	0.5867	0.6031	0.6303	0.6608	0.6940	0.7272	0.7562	0.7895	0.8185	0.8556
50	0.5591	0.5709	0.5942	0.6252	0.6545	0.6886	0.7242	0.7537	0.7883	0.8166	0.8556
75	0.5503	0.5651	0.5911	0.6216	0.6554	0.6890	0.7272	0.7535	0.7893	0.8176	0.8556
100	0.5419	0.5629	0.5853	0.6165	0.6515	0.6880	0.7232	0.7543	0.7884	0.8177	0.8556

Megállapítottam, hogy a szomszédokszám növelése javulást okozott a rendszerben. Kis szomszédokszám esetén érdemes volt ötvözni a CB és CF technikákat, mivel drasztikus javulást jelentett a hibrid technika. Ahogy az alfa értéke közelít az egyhez - ami a tiszta tartalom alapú szűrés esete, a szomszédokszámok változása egyre kevésbé befolyásolja a predikció pontosságát, mivel a CB független a szülők számától. Emiatt a tulajdonság végett az utolsó oszlop értéke állandó. A táblázatból kiolvasható az is, hogy nagyjából 20 szomszédokszámnál több szomszéd bevonásával a rendszerbe a hibrid módszer már kevésbé pontos predikciót eredményez, mint a tiszta kollaboratív technika.

Az optimális alfa alakulásának szemléltetéséhez elkészítettem a 3.4. ábra, ami az átlagos abszolút hiba értékének alakulását mutatja az alfa paraméter értékének függvényében. Minden vizsgált szomszédokszám esetén felrajzoltam a függvényt. Erről az ábráról könnyedén leolvashatóak az előbb ismertetett következtetések.



3.4. ábra Optimális alfa alakulása

Az ábrán még jobban látszik, hogy 1 szomszéd esetén a hibridizáció drasztikus javulást okozott. A szomszédszámok növelésével egyre kisebb lesz a törés a görbén. 20 szomszédszám esetén közel vízszintes a görbe 0.1-es értékű alfáig, de 20 szomszéd felett a hozzá tartozó függvény végig növekvő.

3.6 Kísérlet eredményei

Dolgozatom ezen részében összefoglalom a korábban ismertetett kísérletek eredményeit. Ezen kívül az (Campos L. M., Fernández-Luna, Huete, & Rueda-Morales, 2010) irodalomban ismertetett eredményeket is bemutatom, majd ezen eredményeket hasonlítom össze a saját ajánlórendszerem eredményeivel.

Megállapítottam, hogy Bayes esetén a számítási idő egy-két nagyságrenddel nagyobb, viszont mivel a becslés nem időkritikus, ezért alkalmazható ajánlórendszerek esetén a Bayes technika.

Összehasonlítottam az általános technikákat a Bayes alapú módszerekkel. BN esetén ezen vizsgálatkor a becslő pontszámot két féle módon választottam ki, első esetben a mediánértékhez tartozó pontszámot, második esetben pedig a legvalószínűbb pontszámot vettem a jövőbeli értékelésnek. Összehasonlítás első része az átlagos abszolút hiba vizsgálata volt ((1.5) egyenlet alapján). Tartalom alapú előrejelzés esetén kis eltérések voltak a vizsgált technikák között. Legjobbnak a mediánértékhez tartozó pontszám kiválasztása bizonyult, ennél kissé rosszabb eredményt hozott az általános technika, majd pedig a legvalószínűbb érték kiválasztása az általános módszernél kicsit pontatlanabb volt. Kollaboratív szűrés esetén egyértelműen jobbnak bizonyult a Bayes módszer, azon belül pedig a legvalószínűbb

pontszám kiválasztása adott pontosabb predikciót. Összehasonlítás következő lépéseként az F-mértéket vizsgáltam ((1.7) képlet alapján). F-mérték esetén a top-N halmaz a legjobb átlagpontszámú filmek felső 5%-át tartalmazta. F-mérték vizsgálat kimutatta, hogy a mediánértékhez tartozó pontszám kiválasztása nem bizonyul jó döntésnek, az F-mérték értéke rendkívül kicsi volt, közelítőleg 0 volt minden esetben.

Következő lépésben a Bayes technikára alapuló kollaboratív szűrés és tartalom alapú előrejelzés eredményeinek korrelációját vizsgáltam. Azt vizsgáltam, hogy a szomszédszámok változtatása milyen irányba befolyásolja e két technika korrelációját. A szomszédság növelésével a CB és CF közötti korreláció is növekedett, de nem teljesen korreláltak egymással. Ez a felismerés is megerősítette annak a lehetőségét, hogy e két technikát ötvözzem.

Hibrid módszer kialakításakor feltártam különböző paraméterek hatását az előrejelzés pontosságára. Változtattam a szomszédszámot, ami által a hibrid rendszer pontossága nőtt. Vizsgáltam az alfa paraméter változtatásának hatását is, ami megadja a hibrid modell két komponensének hozzájárulását a végső pontszámhoz. Megállapítottam, hogy kis szomszédszám esetén a két technika fúziója javulást jelent a két komponens pontosságához képest. Viszont 20 feletti szomszédszám esetén a tisztán kollaboratív szűrés már jobb becslést eredményezett. Saját döntés kérdése, hogy vagy kollaboratív szűrést használva a szomszédszámot növelve javítjuk a becslést, ami nagyobb időigénnyel rendelkezik, vagy pedig kis szomszédszám bevonásával integráljuk a két modellt, így kevesebb idő alatt meglesz az eredmény, és pontosabb lesz a predikció, mint ugyan ezen szomszédszám esetén a tiszta technikák esetén.

A (Campos L. M., Fernández-Luna, Huete, & Rueda-Morales, 2010) irodalom a pontszámok előrejelzésekor a mediánértékhez tartozó pontszám szerepel becsült pontszámként.

3.7. táblázat Irodalmi modell MAE értékei

NS	Hybrid		CF
	EQ	RF	
10	0.7293	0.7360	0.7579
20	0.7307	0.7393	0.7637
30	0.7330	0.7422	0.7681
50	0.7364	0.7467	0.7735
75	0.7401	0.7505	0.7784
CB	0.7837	0.7892	

A tartalom alapú szűrést használó ajánlórendszerre fókuszálva (3.7. táblázat utolsó sora) látható, hogy a becslés rosszabb lett amikor a többet felhasznált műfajokat magasabb súllyal számították. A kollaboratív szűrés eredményét a 3.7. táblázat utolsó oszlopa mutatja. A szomszédok számának növekedésével romlott a predikció pontossága. A sok szülő zajt okozott a modellben, így torzultak a szavazati értékek. A 3.7. táblázat középső részén helyezkedik el a hibrid ajánlórendszer adatai. A hibrid eredményei 6-8%-kal jobbak, mint a CF vagy mint a CB. Ebben az esetben is fennáll, hogy a sok szülő rontja az előrejelzést, illetve ha azonos súllyal vettek minden műfajt jobb eredmény adódott.

A saját eredményeim hasonló formában a cikkbelihez a következő 3.8. táblázatban szemléltetem. Látható, hogy a tartalom alapú előrejelzéskor én egy kicsit rosszabb eredményt kaptam, viszont a többi esetben jelentősen jobbak a kapott értékeim. Ennek oka lehet, hogy (Campos L. M., Fernández-Luna, Huete, & Rueda-Morales, 2010) irodalommal szemben én egy másik következtetési módszert alkalmaztam, miszerint a legvalószínűbb pontszámot választottam ki az előrejelzés értékének. A táblázat hibrid részében az alfa értékét 0.1-re választottam.

3.8. táblázat Saját modell MAE értékei

NS	Hybrid		CF
	EQ	RF	
10	0.6110	0.6123	0.6301
20	0.5923	0.5931	0.5932
30	0.5867	0.5863	0.5768
50	0.5709	0.5706	0.5591
75	0.5651	0.5674	0.5503
CB	0.8556	0.8638	

A cikkel ellenkező megállapításra jutattam, mivel nálam a szomszédszámok növelése javította a predikciót. Az én ajánlórendszerem pontosabb eredményeket produkált kollaboratív szűrés és hibrid módszer esetén is.

4. Összegzés, jövőkép

A vásárló központú marketing elterjedésével az ajánlórendszerek egyre nagyobb jelentőséggel bírnak. Az adatbányászat e fiatal területe egyre szélesebb teret hódít. Az információ túlterhelés és az állandóan változó felhasználói preferenciák ellenére az ajánlórendszerek olyan tartalmakat képesek ajánlani, amik nagy valószínűséggel tetszeni fog az adott célfelhasználónak. Az ajánlórendszerek fontossága miatt dolgozatomban e rendszerekkel foglalkoztam. Kutatásom során megismerkedtem a Bayes alapú adatstruktúrával, ajánlási algoritmusokkal. A megszerzett ismereteknek köszönhetően MATLAB-ot használva programot készítettem olyan függvényekkel, melyekkel felépítettem a Bayes hálóra támaszkodó kollaboratív szűrést alkalmazó és tartalom alapú ajánlást alkalmazó ajánlórendszereket, valamint ezeket kombináló hibrid ajánlórendszert. Ezek mellett elkészítettem az általános ajánlási algoritmusok modelljét is. Összehasonlítottam a módszereket. Vizsgáltam a számítási időt kollaboratív ajánlások esetén, és arra a megállapításra jutottam, hogy a szomszédszám növekedésével a számítási idő is nő. Részletes számítást nem végeztem, de hozzávetőlegesen $n \log n$ -es számítási idejű. Mivel az ajánlórendszerek nem időkritikus, így a nagyobb számítási idővel rendelkező Bayes technika is kivitelezhető. Az előrejelző algoritmusomban a pontszám előrejelzésének a legvalószínűbb érték kiválasztását javaslom. Kimutattam, hogy a Bayes alapú technikák jobb eredményre vezetnek. Vizsgáltam a módszerek paramétereinek hatását a predikció pontosságára.

Megállapítottam, hogy kollaboratív szűrés esetén a hasonló felhasználók számának növekedése pontosabb előrejelzést okozott; tartalom alapú előrejelzésnél pedig ha minden műfajt azonos súlyúnak vettem, akkor jobb eredményt kaptam mint ha súlyoztam volna műfajokat az alapján, hogy hány filmet soroltak az adott műfajba. Azt az eredményt kaptam, hogy a tartalom alapú és a kollaboratív szűrés eredményei nem teljesen korrelálnak, így megerősítette bennem annak ötletét, hogy a hibrid technika javíthatja az előrejelzés pontosságát. Hibrid rendszer elkészítéséhez a tartalom alapú ajánlás és a kollaboratív szűrés arányát több esetben is megvizsgáltam. Arra a megállapításra jutottam, hogy kis szomszédszám esetén a hibrid modell pontosabb, mint a két komponense külön-külön. Nagy szomszédszám esetén a kollaboratív szűrés bizonyult jobbnak. Későbbiekben célom hogy az ajánlórendszert továbbfejlesztsem, hogy még pontosabb predikciót eredményezzen. Ezen felül célul tűztem ki a továbbiakban, hogy más ajánlási és előrejelző módszereket is tanulmányozzak, felhasználjak, hogy az ajánlórendszer előrejelzésének, ajánlásának minősége javuljon. Tervezem a kollaboratív és a tartalom alapú ajánlás fúziójának továbbfejlesztését, további hibrid technika generálását.

Köszönettel tartozom témavezetőmnek, Dr. Abonyi Jánosnak a dolgozat elkészítésében nyújtott segítségéért, rendkívül értékes tanácsaiért és türelméért.

Függelék

Változók a programban

- **D:** megadja, hogy mely terméket mely tulajdonság(ok) ír(nak) le.
- **R:** azt tárolja, hogy mely felhasználó mely terméket osztályozta.
- **Y:** magukat a pontozásokat tárolja
- **Y_test:** Y-hoz tartozó 20%-os teszt halmaz
- **H:** mely felhasználó mely felhasználóhoz milyen mértékben hasonlít
- **S:** mely film mely filmhez hasonlít
- **nf:** műfajok száma
- **nu:** felhasználók száma
- **nm:** filmek száma
- **nk:** melyik műfaj hányszor lett használva a filmek leírásához
- **F:** műfajokhoz tartozó prior valószínűségeket tárolja, ha minden műfaj különböző súlyú (RF)
- **F2:** műfajokhoz tartozó prior valószínűségeket tárolja, ha minden műfaj azonos súlyú (EQ)
- **M:** normalizáló faktor
- **Uparent:** megadja, hogy melyik felhasználónak hány szülője van
- **alfa:** hibrid rendszernél a két módszer súlyozása
- **Wi:** az I csomópontokhoz tartozó súlyokat tárolja
- **Wu:** az U csomópontokhoz tartozó súlyokat tárolja
- **Wv:** a V csomópontokhoz tartozó súlyokat tárolja
- **N1:** Azoknak a filmeknek a száma, amit mindkét felhasználó értékelt, az első s-re, a másik t-re értékelt.
- **N2:** Azoknak a filmeknek a száma, amit mindkét felhasználó értékelt, a második t-re
- **MAE:** átlagos abszolút hiba
- **F_measure:** F mérték értéke
- **Prec:** pontosság
- **Rec:** fedés
- **Y_collab:** Bayes módszer, kollaboratív szűrés becsült Y
- **Y_contbase:** Bayes módszer, tartalom alapú szűrés becsült Y

- **Y_hibrid**: Bayes módszer, hibrid becslő Y
- **Y_collab_alap**: általános módszer, kollaboratív szűrés becslő Y
- **Y_contbase_alap**: általános módszer, tartalom alapú szűrés becslő Y

Fájlok-függvények a programhoz

- Adatbetöltő függvények
 - *ratingMatrix.m* – szavazatok betöltése $\rightarrow Y, R$ mátrix létrehozása
 - *moviesMatrix.m* – filmek besorolásának betöltése $\rightarrow D$ mátrix létrehozása
 - *testMatrix.m* – teszhalmaz betöltése $\rightarrow Y_{test}$ mátrix létrehozása
- Adat-előkészítő függvények
 - *hasonlo.m* – felhasználók hasonlóságának számítása. Két paramétere R és $Y \rightarrow H$ mátrix létrehozása
 - *simitem.m* – filmek hasonlóságának számítása. Paramétere $D \rightarrow S$ mátrix létrehozása
 - *sulyok.m* – minden szinthez (műfaj, film, felhasználó, kollaboratív) tartozó súlyok kiszámítása. Paraméterei: $D, Y, H, R \rightarrow F, F2, Wi, Wu, Wv$ mátrixok létrehozása
- Előrejelzések
 - *collab.m* – Bayes alapú kollaboratív szűrés alkalmazása, paraméterei a H, Y, Wv , szomszédok száma $\rightarrow Y_{collab}$ mátrix létrehozása
 - *contbase.m* – Bayes alapú tartalom alapú szűrés alkalmazása, paramétere az $F, Wi, Wu, D \rightarrow Y_{contbase}$ mátrix létrehozása
 - *hibrid.m* – előző kettő Bayes alapú technika ötvözése, paraméterei a H, Y, F, D, Wi, Wu, Wv , szomszédok száma, $\alpha \rightarrow Y_{hibrid}$ mátrix létrehozása
 - *alap_collab.m* – általános kollaboratív szűrés, paramétere a H, Y , szomszédok száma $\rightarrow Y_{col_alap}$ mátrix létrehozása
 - *alap_contbase.m* – általános tartalom alapú szűrés, paraméterei az $Y, D \rightarrow Y_{contbase_alap}$ mátrix létrehozása
- Kiértékelések
 - *mae.m* – MAE számítása, paramétere teszt halmaz és a becslő $Y \rightarrow MAE$ érték megjelenítése
 - *PrecRec.m* – F mérték számítása, top-N-ben az átlagosan legjobbra értékelt felső 5% film szerepel, paramétere teszt halmaz és a becslő $Y \rightarrow F_mérték$ értékének megjelenítése

Irodalomjegyzék

- [1] Abonyi, J. (2006). In *Adatbányászat a hatékonyság eszköze* (p. 367.). Budapest: ComputerBooks.
- [2] Campos, L. M., Fernández-Luna, J. M., & Huete, J. F. (2005). A Bayesian Network approach to Hybrid Recommending System. Spain.
- [3] Campos, L. M., Fernández-Luna, J. M., Huete, J. F., & Rueda-Morales, M. A. (2010). Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks. In T. Denoeux, *International Journal of Approximate Reasoning* 51 (pp. 785–799). Barcelona: Elsevier.
- [4] Dietmar, J. (2011). *Tutorial: Recommender Systems*. Barcelona.
- [5] Engedy, B. (2010). *Ajánlórendszerek*. Budapesti Műszaki és Gazdaságtudományi Egyetem, Budapest.
- [6] Fidy Judit dr., M. G. (2005). Biostatisztika című könyv. InforMed 2002 Kft.
- [7] Gravity, R. (2012). *Gravity R&D*. Retrieved október 20, 2013, from <http://www.gravityrd.com/>
- [8] GroupLens. (2003). *MovieLens 100k*. Retrieved augusztus 30, 2013, from <http://www.grouplens.org/datasets/movielens/>
- [9] Hahsler, M. (2011). *A Framework for Developing and Testing, Recommendation Algorithms*.
- [10] Koller, D. (2012). *Probabilistic Graphical Models*. Coursera, Stanford University.
- [11] Koller, D., Friedman, N., Getoor, L., & Taskar, B. (2007). Graphical Models in a Nutshell. In *Introduction to Statistical Relational Learning*. MIT Press.
- [12] Konstan, J. A. (2013). *Introduction to Recommender Systems*. Coursera, University of Minnesota.
- [13] Liebowitz, E. J. (2008). Application of data mining techniques in customer relationship management: A literature review and classification. In L. X. E.W.T. Ngai, *Expert Systems with Applications* 36 (pp. 2592–2602). ELSEVIER.
- [14] Ng, A. (2012). *Machine Learning*. Coursera, Stanford university.
- [15] Olson, D. L., & and Delen, D. (2008). *Advanced Data Mining Techniques*, Springer, 1st edition.
- [16] Peppers, D., & Rogers, M. (2004). *Managing Customer Relationships*. Wiley.
- [17] Ricci, F., Rokach, L., Shapira, B., & Kantor, P. B. (2011). *Recommender System handbook*. London: Springer.

- [18] Russell, S., & Norvig, P. (2005). Mesterséges intelligencia elektronikus modern megközelítésben, Második, átdolgozott, bővített kiadás. Budapest: Hungarian Translation Panem.
- [19] Smita, & Akshita. (2013). Article: Recommender System: Review. New York: Foundation of Computer Science.
- [20] Tikk, D. (2011). *Mire jók az ajánlórendszerek?* Retrieved from http://barcamp6.ap.hu/?tPath=/view&application=freedocumentview&freedocumentview_type=save&freedocumentview_site=6&freedocumentview_id=17