

TDK Dolgozat

Készítette:
Rechner István

PÁZMÁNY PÉTER KATOLIKUS EGYETEM
INFORMÁCIÓS TECHNOLÓGIAI KAR
SZOFTVER- ÉS NYELVTECHNOLÓGIA



Ajánlórendszer modern megközelítésben

ajánlási algoritmus fejlesztése nagy felhasználószámú tartalomszolgáltató
rendszerek valós közönségei számára

c. TDK Dolgozat

Készítette:
Rechner István
4. évfolyam

Témavezetők:
dr. Lukács Gergely egyetemi docens,
dr. Oláh András egyetemi docens

BUDAPEST

2012

TARTALOMJEGYZÉK

Tartalomjegyzék.....	iii
Ábrajegyzék	v
Táblázatjegyzék	vi
1 Bevezetés	1
1.1 Motiváció.....	1
1.2 Célkitűzés	5
1.3 A dokumentum felépítése.....	6
2 Ajánlórendszerek a modern informatikában.....	7
2.1 A modern ajánlórendszerek szerepköre.....	7
2.2 Adatforrások az ajánlórendszerben.....	9
2.3 Ajánlórendszerek mérőszámai.....	10
3 Klasszikus ajánlási módszerek.....	13
3.1 Tartalom alapú ajánlás.....	13
3.2 Kollaboratív szűrés	19
3.3 Kontextus alapú ajánlás	22
4 Ajánlás közösségek részére.....	25
4.1 A közösségi ajánlás, mint csoportos döntéshozási feladat	25
4.2 Az állapotátmeneteken alapuló vélemények aggregálása.....	27
4.3 Közösségek, mint a szolgáltatóipar valós közönségei	30
5 Algoritmikus módszerek, alkalmazott metrikák	34
5.1 Az ajánlási folyamat áttekintése	34
5.2 Rendelkezésre álló adatok, ajánlási attribútumok	37
5.3 Reprezentáció, az alkalmazott tárolási módszer.....	38
5.4 Használt reprezentáció, alkalmazott metrikák.....	42
5.5 Előszűrési feladatok.....	46
5.6 Algoritmikai feladatok, alkalmazott megoldások.....	47
5.7 Utószűrési feladatok	50
5.8 Algoritmikus módszerek: ajánlás a tervezett rendszerben.....	52
6 A megvalósított modell és eredményei	55
6.1 Rendszeranalízis	55

6.2	A tesztkörnyezet	57
6.3	Tesztadatok, tesztesetek és eredmények	57
7	Konklúzió, további feladatok	64
7.1	A célkitűzések verifikálása: a dolgozat eredményei	64
7.2	A felmerülő további feladatok megoldásai	65
7.3	További feladatok, bővítési lehetőségek	67
7.4	Összefoglalás	68
	Idézett forrásmunkák	69

ÁBRAJEGYZÉK

1. ábra: Hosszú Farok.....	3
2. ábra: Felhasználói információk típusai	9
3. ábra: Tartalom alapú információk típusai	9
4. ábra: Közösségi információk típusai.....	9
5. ábra: A tartalom alapú ajánlás általános folyamata	14
6. ábra: A klasszikus csoportos döntéshozás folyamata	31
7. ábra: Csoportos döntéshozás folyamata a szolgáltatóiparban.....	31
8. ábra: Az ajánlási folyamat fő fázisai.....	34
9. ábra: Az értékelések reprezentálása ritkás (sparse) mátrixként	39
10. ábra: Gráf alapú reprezentáció: a színek a különböző típusú csomópontokat jelölik ..	40
11. ábra: Hipergráf alapú reprezentáció: a különböző színű csomópontok a különböző típusú tartalmakat jelölik	41
12. ábra: Előrecsatolt neurális hálózat	44
13. ábra: Zeneszámok közti hasonlóság megállapítása.....	48
14. ábra: Ajánlás individuális felhasználók számára	48
15. ábra: Közösségi ajánlás folyamata.....	49
16. ábra: Felhasználók eloszlása a 16-45 éves korosztályban	50
17. ábra: A prototipikus modell komponensdiagramja.....	56

TÁBLÁZATJEGYZÉK

1. táblázat: Részlet az Apache Mahout hasonlóságszámításaiból.....	59
2. táblázat: Részlet az Apache Mahout mátrixfaktorzással történő predikciójából	60

ÁTTEKINTÉS

Az elmúlt évek folyamata az információs technológiai iparban az internet és a mobil eszközök rohamos terjedését eredményezte. A tartalomszolgáltató rendszerek megjelenésével igény keletkezett olyan jól skálázódó rendszerekre, melyek lehetővé teszik a tartalmak személyre szabását, és ezzel megoldást jelenthetnek az információátöltődés problémáira. Az ajánlórendszerek szerepe egy ilyen rendszerben felbecsülhetetlen, az alkalmazásuk eredménye az eladási mutatók és a felhasználói élmény ugrásszerű növekedése. Ez napjaink meghatározó e-kereskedelmi és web-portáljai mellett kiemelten fontos a zenei és más médiatartalmat szolgáltató rendszerek esetében is.

A mobilinternet és az okostelefonok elterjedése sokak számára tette elérhetővé a közösségi portálok folyamatos, nagymértékű használatát, így kialakulóban vannak olyan közösségi szolgáltatások, melyek egy adott, valós közönségre koncentrálnak. Ez a terület még kiforratlan, a tartalomszolgáltató rendszereknek pedig hatalmas piacot jelent.

A klasszikus ajánlási módszerek arra keresik a megoldást, hogy hogyan lehet egy-egy termék és felhasználó közti hasznosságot maximalizálni, azonban ez a feladat átalakulóban van. A modern rendszerekben már nem feltétlenül egy-egy felhasználónak, hanem felhasználók egy-egy csoportjának kell megfelelő tartalmakat találni, ráadásul a klasszikus rendszerek figyelmen kívül hagyják az ajánlási lista elemeinek egymásra gyakorolt hatásait, mely nagyban befolyásolhatja azok hasznosságát.

A dolgozat elsődleges célkitűzése egy olyan zenei tartalmak ajánlására szakosodott, jól skálázódó prototipikus modell elkészítése, mely képes sok millió felhasználónak folyamatos, személyre szabott, internetes rádiószolgáltatást nyújtani a felhasználók és a tartalmak egymás közti relációinak, interakcióinak algoritmikus elemzésével és feldolgozásával.

A rendszer a megoldásra felhasználja a mesterséges intelligencia és az adatbányászat kapcsolódó módszereit, és egy olyan gráf alapú megoldást dolgoz ki, mely az adott cél kiszolgálására – az erőforrások és az igények figyelembevételével – alkalmasabb lehet, mint a klasszikus módszerek.

A dolgozatban bemutatott modellt a gyakorlatban alkalmazva kiszolgálhatóak a szórakozóhelyek valós közönségei, mely növeli a felhasználói elégedettséget és a profitot. Emellett a rendszer megvalósításának köszönhetően a Hosszú Farok görbéje jelentősen átalakulhat, felemelkedést jelentve több olyan előadó számára, akik nehezen tudnak betörni a zeneiparba.

ABSTRACT

During the last couple of years the development in the information technology industry lead to the rapid expansion of internet and mobil devices. By the appearance of the content providing systems there is a growing need for scalable systems that solve the problem of information overload by personalizing contents. In such a set-up the role of recommendation systems is invaluable. The result of their application is a leaping growth in the sales scales and user experience. Besides today's determining e-trade business and web pages, they are significantly important in the case of music and other media content providing systems.

As the expansion of mobile networking and smartphones made it possible for many users the continuous and large scale use of social networks, there are forming community services that concentrate on a given real audience. This field is immature and content providing systems have a vast market.

Classical recommendation methods are seeking for a solution for the maximalization of usefulness between a product and the user, however, this function is in the process of transformation. Modern systems do not necessarily need to find adequate contents for each user but for groups of users, moreover, classical systems ignore the interference between the elements of recommendation lists and that could strongly influence their usefulness.

The major aim of the study is to construct a well scalable prototypic model specialized in recommending musical content, that is capable of providing continuous, personalized internet radio service for millions of users by algorithmically analysing and processing relations and interactions between the contents and the users.

The system uses related methods of artificial intelligence and data mining and develops a graph-based solution, in regard to resources and needs, that is more adequate for serving the given purpose than classical methods.

The practical use of the model presented in the study could serve the real audience of clubs and could increase user satisfaction and profit. In addition, owing to the realization of the system the Long Tail curve can significantly transform and give the opportunity to uprise for many artists who have difficulties with breaking into the music industry.

1 Bevezetés

1.1 Motiváció

Bár az ajánlórendszerek kialakulása az 1990-es évekig visszavezethető, megfigyelhető, hogy kutatásuk és fejlesztésük csak az utóbbi 5-10 évben lett igen aktív téma. Az elmúlt években egyre nagyobb szerepet kapnak azok a megoldások, melyek kifejezetten egy-egy média-tartalomra (például filmekre, cikkekre, zeneszámokra) koncentrálnak. A téma egyre fontosabb, hiszen egyre nagyobb szükség van e rendszerek fejlesztésére és átalakítására, hogy képesek legyenek még több fogyasztót kiszolgálni úgy, hogy az ajánlás megfelelően pontos legyen.

Annak ellenére, hogy az ajánlási módszerek fejlődése nagyrészt a tartalomszolgáltatás folyamatos átalakulásának köszönhető, megfigyelhető, hogy a tartalomfogyasztás is nagymértékben változik, és ez kiemelten igaz a zenei tartalmakra. A mobil eszközök és a nagysebességű internet nagymértékű terjedésének köszönhetően a felhasználók hatalmas zenei kollekciókkal rendelkeznek, melyekből azonban nagyon kevés zeneszámot hallgatnak rendszeresen, sőt, a zeneszámok egy része lejátszásra sem kerül. Egyre inkább előtérbe kerül tehát az a probléma, hogy a rengeteg tartalom között a felhasználók nem tudnak megfelelően kiigazodni. Az 5000 iPod-felhasználó részvételével készült felmérés [1] kimutatta, hogy a lejátszások 80%-át a rendelkezésre álló zeneszámoknak csupán 23%-a adja, míg az említett tartalmak 64%-át soha sem hallgatták meg a felhasználók. A probléma mértéke ráadásul egyre növekszik, hiszen egyre több zeneszám készül egyre több előadó által minden évben. 2008-ban az Egyesült Államok jelenlegi legnagyobb online zeneboltja, az iTunes 4 millió új zeneszámot kategorizált, mely nagyjából 400.000 albumnyit tesz ki. Manapság szinte minden előadónak van honlapja (ezt nagyban segítette a MySpace, majd a Facebook és a Youtube), így már szinte minden zeneszám ingyen elérhető online, ráadásul a rengeteg mobil eszköznek köszönhetően a felhasználók is egyre több időt töltenek zenehallgatással, továbbá megállapítható, hogy a zenei tartalmak

egyre jobban beleivódnak más médiákba is, például a televíziós riportokba, vagy reklámokba [2].

Ennek ellenére megfigyelhető, hogy a zenei tartalomfogyasztás csupán néhány előadó felé irányul [3]. A zenei tartalmak megfelelő szűrésével, felfedezésével és személyre szabott ajánlásával ez átalakítható lenne, és a kevésbé ismert, amatőr előadók, kezdő zenekarok is érvényesülni tudnának.

1.1.1 Kutatási motiváció

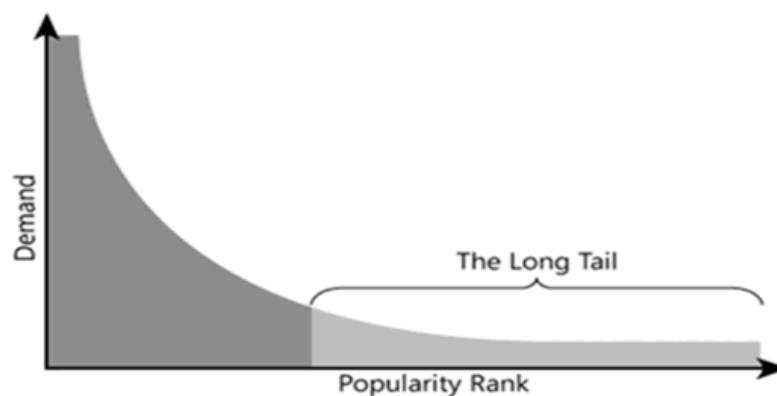
A legnagyobb zenei információkinyeréssel foglalkozó ISMIR (The International Society For Music Information Retrieval) adataira alapozva megállapítható, hogy az ajánlórendszerek, főként a zenei ajánlórendszerek iránti érdeklődés rohamosan nő. A kutatásoknak két nagyon fontos iránya van: egyrészt a jelenleg már ismert, klasszikus ajánlási algoritmusok (például a kollaboratív szűrés, vagy a demográfiai alapú ajánlás) továbbfejlesztése, másrészt a tartalmak és a felhasználók közti hasonlóság minél pontosabb meghatározása az információ kinyerésének minél optimálisabb megvalósításával.

Ahogy az látható, az eddigi módszerek – bár foglalkoznak személyre szabott ajánlással – csak elnagyolva, vagy egyáltalán nem foglalkoznak a közösségi, csoportos ajánlással, mint megoldandó feladattal. Az ajánlórendszerek emellett az egyik leginkább multidiszciplináris terület, hiszen olyan területek határán helyezkedik el, mint a mesterséges intelligencia és gépi tanulás, az adatbányászat és az információ kinyerés, valamint az emberi és gépi interakciót vizsgáló tudományok.

A modern rendszerek emellett felvetnek még egy óriási mértékű problémát: a rohamosan növvő felhasználó és tartalomszám megnöveli a klasszikus megoldások időszükségletét, így lassan ellehetetlenedeg egy-egy módszer alkalmazása a nagy felhasználószámú tartalomszolgáltató rendszerek esetén. Emellett megfigyelhető, hogy a modern rendszerek különféle szolgáltatásai miatt nem olyan egyértelmű egy-egy elem felhasználóra tekintett hasznossága, mint régebben.

1.1.2 Gazdasági motiváció

A tartalomfogyasztási szokások, főként a zenei tartalmakra tekintve radikális változásokon estek át. Megfigyelhető lett egyfajta természetes igény a toplisták követése iránt. A hagyományos és az internetes rádiók is csupán néhány slágert játszanak nap, mint nap, melyek egymást váltják, rendkívül rövid ideig népszerűek. Az 50 legkelendőbb zenei album nagy részét a 70-es, 80-as években adták ki, 2000 óta pedig egyetlen lemeznek sem sikerült a listába bekerülnie: megfigyelhető tehát a slágerek iránti felfokozott igény, de ez mégsem rendelkezik gazdasági húzóerővel. [3] A jelenlegi ajánlási módszerek arra törekednek, hogy minél pontosabban megjósolják, hogy mit *szeretnének* a felhasználók, azonban ennek hátrányaként ezek a rendszerek szinte csak a legnépszerűbb, jól ismert tartalmakat ajánlják, amely nagyban csökkenti az *ajánlás minőségét*. Mivel piaci körökben az ajánlás célja a minél több eladás, így a jelenlegi ajánlórendszerek a hosszú farokként (Long Tail) ismert görbe első részéből, a népszerű tartalmak közül ajánlanak. Ha azonban a hosszú farok hátsó részére koncentrálunk, látható, hogy a letöltések, és velük együtt az eladások száma nem elhanyagolható még a száz-, sőt a nyolcszázvezredik „helyezettnél” sem, és bár számuk a 800.000-dik zeneszám után már jelentősen csökken, azonban még mindig 16 millió feletti, ami az összes eladásnak több, mint 15%-a. Egyenként a zeneszámok eladásai elhanyagolhatóak, azonban együttesen már a piac lényeges meghatározói. [3]



1. ábra: Hosszú Farok

Megfigyelhető tehát, hogy a ritkábban keresett termékek forgalma együttesen jelentősen túlnövi a slágertermékek összforgalmát. Egy, a későbbiekben részletesen felvázolt (zenei) ajánlórendszer pedig ezt a piacot jelentősen átalakíthatja a görbe

“megemelésével”, mely egyaránt rendkívül hasznos a fogyasztóknak (zenehallgató felhasználók) és a szolgáltatóknak (amatőr és professzionális előadók) is.

A dolgozatban megvalósítandó rendszer célja tehát nemcsak az eladási mutatók növelése, hanem a felhasználók és valós közösségeként formálódó felhasználói csoportok *segítése* abban, hogy számukra minél érdekesebb, hasznosabb tartalmat fedezhessenek fel, valamint az előadók *segítése* abban, hogy az általuk létrehozott, művészi értékekkel rendelkező zenei tartalmakat minél nagyobb célközönséghez minél könnyebben és olcsóbban el tudják juttatni.

1.1.3 Piaci motiváció

A tartalomszolgáltatási ipar hatalmas változásokon ment keresztül az elmúlt években. Korábban a hagyományos boltokban mindent a fizikai korlátok határoztak meg: egy átlagos lemezboltban legalább 4 darabot kellett albumonként eladnia, hogy fenntartható legyen, ami megközelítőleg 1.27 cm-t foglalt el a polcokból. Ugyanez a fizikai korlát figyelhető meg a rádiók területén is: a szolgáltatási spektrum szintén véges, és igen drága, így mindezidáig azok a (sláger) termékek kerültek előtérbe, melyek az eladások nagyobb részét adták. Ez az időszak a „hiány” kora, mely az internet elterjedésének köszönhetően a „bőség” korára változott. A digitális média korában pedig két feladat került előtérbe: mindent elérhetővé tenni, és segíteni a felhasználóknak megtalálni a nekik megfelelő tartalmat [3]. Erre a két feladatra megoldást nyújthat egy megfelelően kidolgozott zenei ajánlórendszer.

Kifejezetten a zeneiparra tekintve egyéb változások is megfigyelhetők, melyek nagyban indokolják egy teljes zenei ajánlórendszer, mint termék létrejöttét: az ipar egyre inkább eltekint a zeneszámok esetleges költségétől, és máshol próbálja megteremteni az ezekből származó hasznót: koncertjegyekből vagy a védjegyekkel ellátott egyéb termékekből (például ruházati árucikkek), esetleg reklámokból. A mai zeneiparra az a legjellemzőbb, hogy a hagyományos zenei kiadók az „ingyenességet” hibáztatják minden problémájukért, azonban nagy hiba lenne összekeverni a zenei kiadókat a teljes zenei piaccal. A zeneszámok eladási mutatóin kívül minden, a piacon szereplő gazdasági mutató rohamosan emelkedik az online letöltések, eladások hatására, és ennek pozitív hatásait minden piaci szereplő érzi. [2] Egy

megfelelően kidolgozott zenei ajánlórendszer (zenei rádiós szolgáltatással) nemcsak ezekre a problémákra nyújtana megoldást, de segíthetné az amatőr előadók minél gyorsabb megismertetését a nagyközönséggel, így segíthetné felfedezni az ezekben rejlő művészeti és gazdasági értéket egyaránt. A tervezett rendszer emellett alkalmas lehetne szórakozóhelyeken a zenei szolgáltatás biztosítására is, növelve azok látogatottságát és ezzel bevételeit.

1.2 Célkitűzés

A dolgozat elsődleges célkitűzése egy olyan, zenei tartalmak ajánlására szakosodott, jól skálázódó, prototipikus modell elkészítése, mely képes a rendszer felhasználóinak folyamatos, személyre szabott, internetes rádiószolgáltatást nyújtani a felhasználók és a tartalmak egymás közti relációinak, interakcióinak algoritmikus elemzésével és feldolgozásával. A megvalósítandó rendszer emellett alkalmas valós szórakozóhelyek közönségét kiszolgálni csoportos, közösségi ajánlás segítségével. Ez elsősorban amatőr előadók műveinek ajánlására koncentrál, azonban a jogi esetek elemzésétől eltekintve alkalmas akár az eddig digitálisan rögzítésre és kiadásra került összes professzionális zenei mű ajánlására is.

E célok eléréséhez egy olyan algoritmus kerül megalkotásra, mely a területen általánosan alkalmazott más módszereket, mint például az (akusztikai) tartalom alapú ajánlás és az értékeléseken alapuló kollaboratív szűrés a zenei tartalmakra, a média-tartalmakra és a közösségi kapcsolatokra, mint újszerű adatforrásra koncentrálna dolgozza tovább, illetve finomítja, egy új megoldásban egyesítve ezek tulajdonságait úgy, hogy azok alkalmazásra kerülhessenek akár sokmilliós felhasználó- és tartalomszámú tartalomszolgáltató rendszerekben is.

1.3 A dokumentum felépítése

A dokumentum 3 nagy részre, ezeken belül 7 fejezetre lett felosztva:

- **Az első rész (1-2. fejezet)** egy bevezető és áttekintő rész, melyben a területen általánosan leírt és a modern rendszerekben felmerült problémák kerülnek leírásra, megvilágítva azokat több kapcsolódó terület (például gazdasági, vagy társadalmi) aspektusaiból is. Ebben a fejezetben kerülnek áttekintésre az alapvető feladatok, melyek megoldása szükséges a célkitűzésben kiírtak eléréséhez. A második fejezetben áttekintésre kerülnek az ajánlórendszerek, mint piaci alkalmazások: megfogalmazásra kerülnek szerepköreik, hatásaik a modern informatikában.
- **A második rész (3-4. fejezet)** egy az irodalmat áttekintő és szintetizáló egység, melyben leírásra kerülnek a klasszikus ajánlási módszerek alkalmazhatóságai speciálisan zenei tartalomra. Vizsgálatra kerülnek ezek általános és zenei tartalomra specifikus hibái, valamint az általuk felállított kihívások és részfeladatok. A (4.) fejezet kifejezetten a közösségi ajánlásra, mint modern megközelítésre koncentrál, leírja az általános ajánlási feladatokhoz hozzátársuló kihívásokat, valamint az irodalom felhasználásával javaslatot tesz egy, a rendszer közösségi szolgáltatását végrehajtó megoldásra.
- **A harmadik rész (5-6. fejezet)** egy a rendszer egyedileg kidolgozott algoritmikus módszereit, metrikáit bevezető és leíró egység, mely saját megközelítéssel oldja meg az előző részekben felvetett feladatokat és kihívásokat. A saját megközelítés összehasonlításra kerül a piacon jelenleg megtalálható egyik legnépszerűbb rendszerrel, az Apache Mahouttal, mely összehasonlítás eredményeképpen megszületik a rendszer általános értékelése.

2 Ajánlórendszerek a modern informatikában

2.1 A modern ajánlórendszerek szerepköre

A modern tartalomszolgáltató rendszerek, online rádiók és szórakozóhelyek rendkívül nagy felhasználószámmal és tartalomszámmal rendelkeznek. Ezekben a rendszerekben azonban a felhasználó sokszor nem tud kiigazodni, nem találja meg a számára érdekes tartalmakat. Kimutatható, hogy azok a tartalomszolgáltató rendszerek sikeresebbek, melyeknél már a kezdőlapra is személyre szabott tartalom kerül, ami folyamatosan frissülve igazodik a felhasználó igényeihez. Ilyenkor a felhasználó nem kerül zsákutcákba, melyekből otthagyná a rendszert, hanem tovább böngészi a számára érdekes tartalmakat, ennél fogva az ajánlórendszerek piaci és gazdasági szerepe hatalmasra nőtt.

2.1.1 Ajánlórendszer, mint piaci eszköz

Az ajánlórendszer és az ajánlási feladat általánosan úgy kerül leírásra, mint egy olyan rendszer, ami a felhasználóknak tartalmakat ajánl. Kicsit közelebbről nézve, a dolgozat szemszögéből ez a rendszer zenei tartalmakat ajánl individuális felhasználóknak vagy közösségeknek, azonban áttekintésre kerül az is, hogy mire képes egy általánosan definiált ajánlórendszer. Az ajánlórendszer, mint általánosan használt megoldás a következő üzleti feladatok megoldására alkalmazható [4]:

- Eladási mutatók növelése
- Minél különfélebb tartalom eladása
- Felhasználói elégedettség növelése
- Felhasználói hűség növelése
- A felhasználó igényeinek jobb megértése

Mint ahogyan az a fentebbi pontokból is jól látható, egy ajánlórendszer pontosan azokon a pontokon segíti az e-kereskedelmi, illetve ma népszerű tartalomszolgáltató rendszereket, amelyek bevételeik elsődleges részét képezik. Ezekből a pontokból kiemelten fontos az eladási mutatók növelése, valamint a felhasználói

elégedettségnek köszönhetően visszatérővé váló felhasználók. Ezeket a pontokat azonban nem triviális feladat teljesíteni, a legtöbb mai nagy tartalomszolgáltató rendszer bevételeinek nagy részét e mutatók növelésébe fekteti. Az ajánlórendszerek a fentebb említett célokat az alapvető ajánlási feladatok teljesítésével érik el.

2.1.2 Alapvető ajánlási feladatok

A klasszikus, formális ajánlási feladat (azaz a predikciós probléma kiterjesztése) mellett számos olyan feladat van, melyet egy ajánlórendszernek meg kell oldania. Az alapvető feladatok a következők: [5] megfelelő tartalmak, azon belül is *minden megfelelő tartalom megtalálása*, szűrési feladatok, főképpen az *ajánlási lista* megalkotása, a felhasználói, tartalmi profilok olyan irányú fejlesztése, hogy *segítse* a felhasználókat a kedvelt tartalmak megtalálásában, de csak olyan mértékben, hogy az *megbízható* legyen számukra. Az alapvető feladatokon kívül a modern rendszereknek köszönhetően megjelennek még olyan feladatok is, mint az *egymás segítése*, mely utal a *közösségi* ajánlásra, valamint a más felhasználók *befolyásolására*, természetesen pozitív értelemben.

2.1.3 Formális definíció

A feladat formalizálva: [6] legyen C az összes felhasználót reprezentáló halmaz, illetve legyen S az összes ajánlható elem halmaza. Ezen halmazok mindegyike milliós elemszámú lehet. Legyen u az a hasznosságfüggvény, mely s elem c felhasználóra való hasznosságát írja le, azaz: $u: C \times S \rightarrow R$, ahol R egy teljesen rendezett halmaz. Továbbá minden $c \in C$ felhasználóra feladat olyan $s' \in S$ elemet választani, mely maximalizálja az adott c felhasználóra a hasznosságát. Összefoglalva tehát:

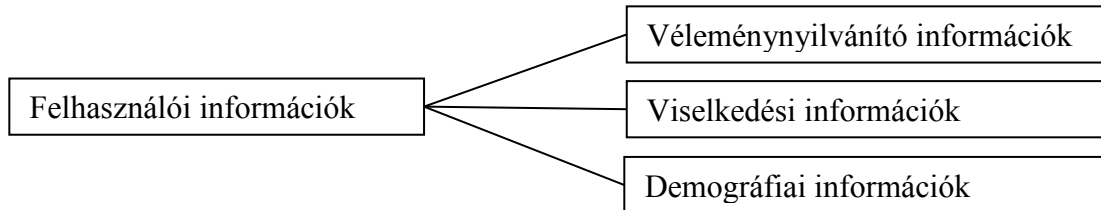
$$\forall c \in C: s'_c = \arg_{s \in S} u(c, s)$$

Az elsődleges feladat tehát az u hasznosságfüggvény értékeinek minél pontosabb becslése, melyhez a felhasználó értékelései szolgálnak alapul. Az értékelések egyszerű $\langle c, s, r \rangle$ hármassokból állnak, ahol c jelenti a felhasználót, s az elemhalmazt, r pedig az értékelést:

$$r_{opt} = u(c, s)$$

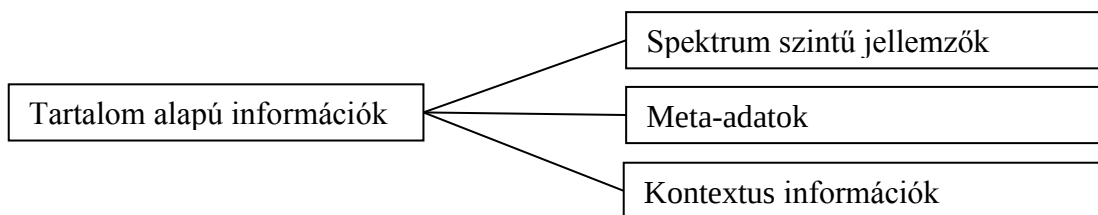
2.2 Adatforrások az ajánlórendszerben

Bármely mesterséges intelligenciával rendelkező rendszer, így egy ajánlórendszer tervezésekor és implementálásakor is a legelső, és egyik legfontosabb feladat meghatározni a használandó adatok típusait. Egy személyre szabott ajánlórendszer feladatából kifolyólag két nagyon fontos információval biztosan rendelkezik: a



2. ábra: Felhasználói információk típusai

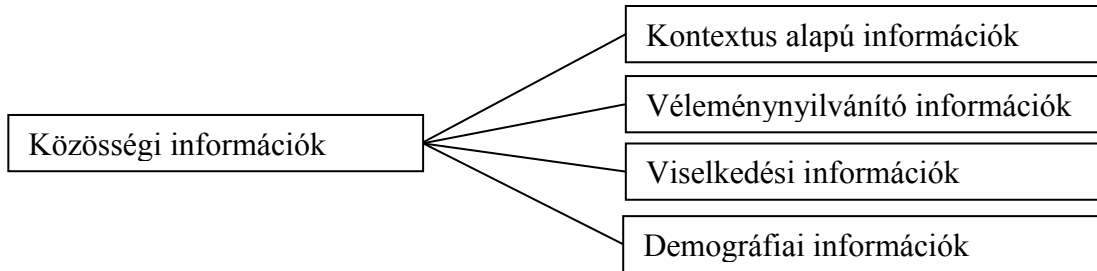
felhasználói információval, valamint az ajánlható tartalmak adataival. Ezek a fő adatkategóriák további alkategóriákra [4] bonthatóak aszerint, hogy milyen módon került kinyerésre az adott információ. A rendszer egy adott felhasználója jellemezhető a *véleményével*, a *viselkedésével*, illetve a *demográfiai adataival*. A tartalomszolgáltató rendszer egy adott eleme pedig jellemezhető jelfeldolgozással *spektrum-szinten*, leírható *meta-adatokkal*, mint jellemző attribútumokkal, valamint adatbányászati módszerekkel megadhatóak a rá jellemző *kontextus-információk*, melyek tulajdonképpen meta-adatok, csupán az adatok kinyerésének módja jelent különbséget.



3. ábra: Tartalom alapú információk típusai

Tartalomszolgáltató rendszerek – főként ajánlórendszereket használó rendszerek – esetén azonban nem csak egy-egy felhasználó individuális információi ismertek,

hanem egy (vagy több) közösségé is. Ezek a közösségek lehetnek akár milliós számúak is (például felfogható a teljes felhasználóbázis is úgy, mint közösségi csoport). Ennek következtében megállapítható a közösségi információ is, mely a kontextus alapú, a véleménynyilvánító, a viselkedési és a demográfiai alkategóriákra oszthatók.



5. ábra: Tartalom alapú információk típusai

2.3 Ajánlórendszerek mérőszámai

Az ajánlási feladatban látató, hogy a cél a felhasználóra vetített hasznosságfüggvény maximalizálása. A predikció pontosságának növelése rendkívül fontos, azonban nagyban számít a postosság mérésének módszere is. A következőkben bemutatásra kerülnek az erre alkalmazható módszerek, melyek alapján később az ajánlórendszer mérése történhet.

2.3.1 Mérési módszerek típusai

Az ajánlórendszer mérésére alkalmazható módszerek az alábbi kategóriákba sorolhatóak: [7]

- **Rendszer központú:** a rendszer központú mérési módszer annak pontosságát méri, hogy az ajánlórendszer mennyire pontosan tudja meghatározni azokat az értékeléseket, melyeket a felhasználó korábban adott. Ez a módszer az explicit értékeléseken alapuló kollaboratív szűrés mérésére alkalmas leginkább.

- **Hálózat központú:** a hálózat központú mérési módszer a tartalmak és a felhasználók közötti hasonlósági topológia mérésére koncentrál. Ennek mérésére komplex hálózati analízist (CNA) használ.
- **Felhasználó központú:** a felhasználó alapú mérési módszer a felhasználóra nézett hasznosságot méri.

2.3.2 Alkalmazható mérési módszerek

A tervezett ajánlórendszerre a mérési módszerek mindhárom típusa alkalmazható. Kifejezetten az algoritmus hibájának mérésére a rendszer központú módszerek szolgálnak, a hasonlósági hálózat (azaz a felhasználók hasonlóságának és a tartalmak hasonlóságának) mérésére jó választás a hálózat központú módszer, míg a központi feladatra megoldásként adott ajánlási lista hasznosságát a felhasználóra nézve a felhasználó központú módszerekkel lehet a legjobban mérni. [7] Éppen ezért a dolgozat olyan egymás mellett futó mérési módokat javasol, melyek egyenként mérik az algoritmus, a hálózat és a hasznosság mértékét. Ezek alapján a hibák alapján az ajánlórendszer tanítható, így egyre pontosabb és értékeesebb ajánlási listát tud adni.

2.3.3 A mérési módszerek áttekintése

A hibamérés leggyakrabban úgy zajlik, hogy a tanítóhalmaz (az értékelések halmaza) két részre oszlik, és az egyik részt tanítóhalmazként használva algoritmikusan becslésre kerül a másik. A használható módszerek [7] a következők:

- **Mean Absolute Error (MAE):** a prediktált és a valós értékek közti hiba mértékét a következőképpen határozza meg:

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{R}_i - R_i|$$

ahol $\hat{R}_i$ a prediktált, R_i pedig a valós érték.

- **Root Mean Squared Error (RMSE):** Az RMSE az MSE (Mean Squared Error) gyöke. Az adott prediktált és valós érték közti hibát a következőképpen méri:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{R}_i - R_i)^2}$$

Az RMSE a legszéleskörűbben alkalmazott hibamérési módszer, sok nagy ajánlórendszer használja.

A fenti két általánosan alkalmazott módszer mellett az ajánlás hibájának mérésére alkalmazható még *döntés-alapú* hibamérés, a nyelvtechnológiában is széleskörűen használt *pontosság* és *fedés*, de akár *Kendell-tau* (τ), vagy a tartalomszolgáltató rendszereknél leggyakrabban alkalmazott *A/B teszt* is.

3 Klasszikus ajánlási módszerek

Ebben a fejezetben bemutatásra kerülnek a klasszikus ajánlási módszerek, úgymint a felhasználói értékeléseken alapuló kollaboratív szűrés vagy a tartalom alapú ajánlás, valamint ezek alkalmazása kifejezetten zenei tartalomra. A dolgozat áttekinti, hogy mely kihívásokat, illetve részfeladatokat kell megoldani a klasszikus módszerek a dolgozatban tervezett rendszerben történő alkalmazásakor.

3.1 Tartalom alapú ajánlás

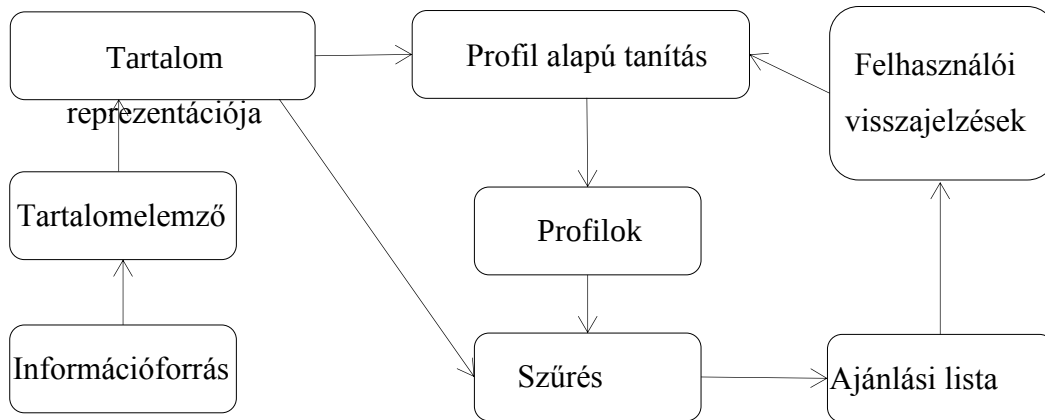
A tartalom alapú ajánlás a legrégebben alkalmazott ajánlási módszer. Az $u(c, s)$ hasznosság függvény s elemre és c felhasználóra nézve az $s_i \in S$ tartalmak $u(c, s_i)$ hasznosságfüggvényből származtatható. Tehát zenei tartalmakra nézve egy s zenei tartalom c felhasználónak való ajánlásakor a tartalom alapú ajánlóalgorithmus elsődleges feladata megtalálni a közös vonásokat azokban az s_i elemekben, melyeket a felhasználó az elmúlt időszakban magasan értékelt. Ezek a közös vonások a zeneszámokat jellemző attribútumok (meta-adatok), melyek az adott zeneszámot jellemzik. Ilyenek lehetnek például a stílus, az előadó, vagy akár specifikusabb jellemzők, mint például hangnem, hangszerelés. A közös vonások mértékét az $S: I \rightarrow R$ hasonlóságfüggvény méri, ahol I jelöli a meta-adatok halmazát. Az ajánlási lista abból a $k \in N$ darab zeneszámból fog állni, melyek hasonlósági mértéke a legnagyobb:

$$\max\{S(s_i, s_{\text{aktuális}})\}$$

Az összes meta-adat halmaza legyen M , az adott elem $s \in S$, a tartalmakat megadó függvény pedig $C: M \rightarrow H$, ahol $H \subseteq M$. Ekkor H az eredményként kapott meta-adatok listája. Az elemek hasonlóságát a meta-adatok hasonlósága adja, mely adatok információkinyerési stratégiákkal érhető el [8].

3.1.1 A tartalom alapú ajánlás klasszikus architektúrája és működése

A tartalom alapú módszernek az ajánlási feladat megoldásához megfelelő információszűrési algoritmusokra van szüksége. Ezen algoritmusokat a rendszerarchitektúrában egy-egy modul reprezentálja [9]:



6. ábra: A tartalom alapú ajánlás általános folyamata

- **Tartalomelemző komponens:** a strukturálatlan információ előfeldolgozási feladatait ellátó modul. Célja, hogy a zenei tartalmat a meta-adatok terébe helyezze valamilyen módon.
- **Tanító komponens:** a felhasználói értékeléseket összegyűjtő és általánosító modul, mely gépi tanulást használ a felhasználói profil megalkotásához. A profilalkotás és az értékelések esetleges becslésének eredményének pontosságáról felhasználói visszajelzés alapján képes tanulni.
- **Információsűrő komponens:** a felhasználói és tartalmi profilokat elemző és összepárosító modul. Ez a modul felelhet az ajánlási lista megalkotásáért is, mely lista egy rendezett, hasonlósági értékek alapján eredményül kapott, zeneszámok azonosítóját tartalmazó lista. Ilyen hasonlósági értékeket adhat például a Manhattan-távolság, a Loglikelihood-hasonlóság, vagy a leggyakrabban alkalmazott Pearson-korreláció.

A tartalom alapú ajánlás lépései [9] alapján a következők:

1. A tartalomelemző komponens információkinyerési technikákkal elemzi a zenei tartalmat (részletesebben a zenei tartalmak tulajdonságainál)
2. Visszajelzések alapján felépül vagy frissül az aktív felhasználó profilja (aktív az a felhasználó, akinek az ajánlási lista éppen készül)
3. A tanító komponens elvégzi a predikciót

A profilépítési szakasz (2.) elvégzéséhez szükség van az u_a felhasználó TR_a tanítóhalmazára. Ez egy $\langle I_k, r_k \rangle$ párokból álló halmaz, ahol r_k az u_a felhasználó által I_k tartalom-reprezentációra adott értékelése. [9] Ha a felhasználói profil elkészült, a tanító komponens a mesterséges intelligencia által alkalmazott gépi tanulási módszerek egyikével, vagy kombinációjával elvégzi a felhasználói- és a tartalomhalmaz összepárosítását.

Az összepárosítás után az ajánlási lista a felső $k \in R$ prediktált értékekből áll össze.

3.1.2 Zenei tartalmak egyedisége

A tartalom alapú ajánlási módszer médiatartalmakra - főként a zenei tartalmakra történő alkalmazásakor fontos megemlíteni azokat a különleges jegyeket, melyek ezek ajánlásakor felmerülő feladatokat eredményeznek, vagy a meglévő feladatok más aspektusú megoldását indokolhatják. Ezek az egyediségek a zeneszámokra még a többi média-tartalommal szemben is igazak: több egyediség merül fel, mint bármely más tartalomnál (például könyvek, filmek).

Az iTunes, az Egyesült Államok legnagyobb zenei viszonteladója 2008-ban 4 millió új zeneszámot (400.000 zenei albumot) jegyzett. Az egyre több zeneszám keletkezésével a fogyasztási mérték is rohamosan nő: az új eszközök megjelenésével radikálisan növekedett a zenehallgatással eltöltött órák száma. [2] A zenei tartalmak száma tehát rohamosan nő, ellenben mind előállítási, mind fogyasztási költségük egyre alacsonyabb. A mobil eszközök és az internet rohamos terjedésének köszönhetően ma már bármely valaha rögzített zeneszám ingyen, néhány kattintással elérhető (Youtube¹, SoundCloud²). Az új, amatőr zenekarokat is támogató startupok

¹ <http://www.youtube.com/>

(pl. BandCamp³) lehetővé teszik, hogy a fogyasztók határozzák meg a tartalmak árát, így elmondható, hogy a zenei tartalom mellett, hogy *olcsó és könnyen elérhető*, jelenleg a *legnagyobb mennyiségben rendelkezésre álló* digitális tartalom. A 2000-es évektől kezdődően, egyre több amatőr előadó művei jutottak el a fogyasztókhoz, és több, merőben új zenei stílus alakult ki és hódított teret (pl. dubstep, djent). Ezek az új stílusok még több amatőr zenekar létrejöttét segítették elő, melynek köszönhetően mára az összes médiatartalmat tekintve a zenei tartalmak *száma a legnagyobb*. Megfigyelhető továbbá, hogy a gyors fogyasztási időnek és annak köszönhetően, hogy nem köti le a fogyasztó teljes figyelmét, a zenei tartalmak *felhasználása a legsokrétűbb*. Alkalmazzák audiovizuális tartalmak (mozifilmek, riportok, reklámok) alatt, rengetegen hallgatják utazás, munka, sportolás vagy tanulás közben. Azonban a filmekkel és egyéb tartalmakkal szembeállítva a zeneszámok *újrafelhasználásának mértéke sokkal nagyobb*, azaz sokkal többször hallgatnak meg újra egy-egy zeneszámot, mint ahányszor egy-egy videót vagy filmet lejátszanak, vagy egy-egy könyvet elolvasnak. Emellett a zenei tartalmaknak van még egy kulcsfontosságú felhasználása: a közösségi rendezvényeken, valamint a *szórakozóhelyeken*.

A zenei tartalmak legfontosabb **egyediségei** más médiákhoz viszonyítva tehát a következők:

- sokkal több áll rendelkezésre, mint más médiatartalmakból
- előállítási költségük jóval kevesebb
- nagymértékű újrafelhasználás
- közösség szervező szerep
- a fogyasztók meghatározhatják a tartalom árát
- jelenlegi legolcsóbb digitális tartalom
- sokrétű felhasználás

² <http://soundcloud.com/>

³ <http://bandcamp.com/>

– könnyen elérhetőek

3.1.3 Zenei tartalmak jellemzése, meta-adatok generálása

A tartalom alapú rendszerekben a tartalmak leírása attribútumokkal, tulajdonságokkal történik. Zenei tartalmak esetén ezek a meta-adatok lehetnek például az előadó, a stílus, a hangnem, stb. Ha több tartalom is ugyanazokkal az attribútumokkal rendelkezik, és a lehetséges attribútumok ismertek, megállapítható, hogy a tartalmak hasonlóak.

Érdekes még megemlíteni más megközelítést is a tartalmak hasonlóságának mérésére: a számítási teljesítmény immár eljutott arra a szintre, hogy véges időn belül megállapítható a zeneszámok közti hasonlóság csupán azok digitális reprezentációinak (digitális adat) elemzésével. Ezek közül a legnépszerűbb módszerek [10] alapján a *blokk szintű jellemzők (Block-Level Features)* kinyerésének módszerei, például a *spektrális mintázatok módszere*, mely azokat a frekvenciakomponenseket modellezi és elemzi, melyek egyidejűleg aktívak. Ezekből átlapolódó blokkok képződnek, melyek dinamikája számít igazán. Az átlapolások miatt az összes blokk helyett azoknak egy része kerül kiválasztásra, majd leírásra, mely blokkcsoportok dinamikájából meghatározható a zeneszámok közti hasonlóság.

3.1.4 Zenei tartalmak által okozott nehézségek

A meta-adatok egyezésének (vagy hasonlósági mértékének) megállapítása nem triviális feladat. A legtöbb rendszerben az attribútumok szöveges jellemzők, nem strukturált, számosított értékek, így a morfológiai különbségek által okozott problémák (mint például a többjelentésű szavak és a szinonimák) [9].

Ezen problémák megoldására a kapcsolódó irodalom [9] szinte mindig szemantikai analízist, valamint szöveg-összehasonlítást (például a Hamming-távolságot, ontológiai elemzést) javasol, azonban a dolgozat erre egy modern, új megközelítésű megoldást ajánl neurális hálózat alkalmazásával a (6.3.4) pont szerint.

Fontos kitérni a meta-adatok összehasonlításánál a morfológiai problémák mellett arra a feladatra is, hogy a zenei ajánlási lista nem tartalmazhat egymás után egyazon

előadótól vagy albumról származó zeneszámokat, mert az nem elégítené ki az eredeti ajánlási feladat feltételeit, miszerint a hasznosságfüggvényt a felhasználók szempontjából maximalizálni kell:

$$\forall c \in C: s'_c = \arg_{s \in S} u(c, s)$$

A zenei hasonlóságot leíró meta-adatok közül a legmeghatározóbbak lehetnek a stílus, a hangszerelés, a zenei játék, valamint az énekes hangja (ha nem instrumentális zeneszámról van szó). Ezek az attribútumok azonos előadó zeneszámainál nagyban megegyeznek, így a tartalom alapú ajánlás ugyanazon előadó számait (helyesen) nagyon hasonlónak gondolná. Ez a matematikai feladat feltételeit kielégíti, azonban a felhasználók számára unalmassá válhat a lista. Ez a probléma előadó-hatásként is ismert.

A leírt helyzet megfigyelhető albumonként, illetve zenei stúdióként is, hiszen ugyanazt az eszközparkot használva, ugyanolyan körülmények mellett, és legtöbbször ugyanazon szakemberek által készül el a felvétel. Ez a probléma album-hatásként is ismert, és szintén amiatt kerülendő, hogy az ajánlási lista ne legyen unalmas. A dolgozat ezen feladatok megoldására elő- és utószűrési megoldást javasol a (6.1.3) pontban.

3.1.5 A tartalom alapú ajánlás általános problémái

A tartalom alapú ajánlás problémái [11] alapján a következők:

- **Kevés információ:** a jelenlegi, klasszikus algoritmusokkal a zenei tartalmak digitális reprezentációiból kinyerhető információk nem elegendőek ahhoz, hogy jelentős sorrend alakulhasson ki a hasonlóságfüggvény eredményei alapján az ajánlási listához. Kevés meta-adat esetén ugyanis nagyon sok ugyanolyan hasonlósági együttható lesz, így nagyon sok zeneszám kerülne be ugyanolyan valószínűséggel az ajánlási listába. Megjegyzendő, hogy ez a probléma nem az elgondolás hibája, hanem az információkinyerési technikák gyengeségének tudható be.
- **Túl nagy hasonlóság:** a tartalom alapú ajánlóalgoritmusok célja, hogy maximalizálják az $S: I \rightarrow R$ hasonlóságfüggvény értékét. Ennek köszönhetően

egymáshoz nagyon közeli elemek kerülnek az ajánlási listába, azonban a hasonló elemek folyamatos ismétlődése és hasonlósága miatt az ajánlórendszer elvesztené egyik fontos feladatát: segíteni a *felfedezést* a hosszú farkok végéről.

- **Hidegindítási probléma:** a tartalom alapú ajánlás pontossága függ attól, hogy mennyi információ áll rendelkezésre egy adott felhasználóról. Ezt az információt a $\langle c, s, r \rangle$ értékelési hármas reprezentálja, melyből új felhasználó esetén nagyon kevés, vagy semmi sem áll rendelkezésre. A hidegindítási probléma a jelenlegi ajánlórendszerek egyik legnagyobb problémája.

3.2 Kollaboratív szűrés

A tartalom alapú ajánlással ellentétben a kollaboratív szűrésnek nincs szüksége a tartalmak jellemzőire, csupán a felhasználók értékeléseire. Ezekből a $\langle c, s, r \rangle$ értékelési hármasokból kerül megállapításra az aktív felhasználóra tekintett, általa még nem értékelt elemek hasznosságfüggvénye. Formálisan: az s elem c felhasználóra vonatkoztatott $u(c, s)$ hasznosságfüggvénye azokból az $\langle c, s, r \rangle$ értékelési hármasokból kerül megállapításra, amit $c_j \in C$ felhasználók s tartalomra adtak. Ezután $R(s): C \times N \rightarrow S \times N$ függvény eredményeként megkaphatóak a felhasználó-hasznosság párok s tartalomra.

Ez a módszer azonban ahhoz vezet, hogy a kollaboratív értékeléseket kihasználva, de tulajdonképpen még mindig a tartalmak hasonlóságán alapul az ajánlás, csupán a hasonlósági heurisztika változik: a digitális reprezentáció és meta-adat hasonlóságok helyett a kollaboratív szűrésnél már csak az értékelések számítanak. A kollaboratív szűrés így ugyanolyan módon alkalmazható bármilyen tartalomra, amiről megállapítható egyfajta felhasználói tartalomértékelési kapcsolat, így a zenei tartalmakra nincs speciális felmerülő feladat.

3.2.1 Mátrixfaktorizálás alapú megoldások kollaboratív szűrésre

A kollaboratív szűrés által leginkább használt algoritmusok az SVD (Singular Value Decomposition) mátrixfaktorizálási algoritmus, és ennek implicit értékelésen alapuló, továbbgondolt változata az SVD++, valamint ennek idő alapú változata, a timeSVD++.

Az SVD megoldások leképezik a felhasználókat és a tartalmakat egy közös látensfaktor térbe, ahol a felhasználó-tartalom interakciók belső szorzatként kerülnek reprezentálásra. Ebben a térben minden i tartalomhoz párosul egy $q_i \in R^f$ vektor, ahol f jelöli a tér dimenzióját. Emellett minden u felhasználóhoz párosul egy $p_u \in R^f$ vektor. Ezek a vektorok a tulajdonképpeni preferenciákat jelölik, értékeik lehetnek negatívak és pozitívak is. [12]

Ezek alapján egy értékelés a következőképpen prediktálható: [12]

1. SVD: $\hat{r}_{ui} = \mu + b_i + b_u + q_i^T p_u$
2. SVD++: $\hat{r}_{ui} = \mu + b_i + b_u + q_i^T (p_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j)$
3. timeSVD++:
$$\hat{r}_{ui} = \mu + b_i(t_{ui}) + b_u(t_{ui}) + q_i^T (p_{u(t_{ui})} + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j)$$

Ahol (b_i, b_u, p_u, q_i) a tanított modell-paraméterek.

Megjegyzés: mivel a dolgozat egy szomszédság alapú, gráf-reprezentált megközelítést vezet be a (6.3.) pontban, ezért a mátrixfaktorizálás további, finom részleteinek részletes leírásától eltekint.

3.2.2 Szomszédság alapú megközelítés

Bár az SVD alapú megoldások pontosabbak is lehetnek a szomszédság alapú megoldásoknál (MAE, RMSE mérése esetén), nagyon fontos megjegyezni, hogy a zenei ajánlórendszer pontosságát nem csak a predikciós pontossága jelzi, hanem annak mértéke is, hogy mennyire segít a felhasználóknak új tartalmakat felfedezni, amiket a rendszer nélkül lehet, hogy sosem ismer meg. A szomszédság alapú megoldás azonban segít ebben, és még számos előnnyel rendelkezik [13].

A módszert szokás két nagy típusra felosztani: a felhasználó-alapú és a tartalom alapú módszerekre.

A **felhasználó alapú** predikciós módszer u felhasználó i tartalomra adott r_{ui} értékelését becsüli meg felhasználva az u -hoz leghasonlóbb felhasználók i -re adott értékeléseit. Az u -hoz leghasonlóbb felhasználók a (felhasználó-alapú) legközelebbi szomszédok [14]. A módszer egyik hátránya, hogy csak azok a felhasználók lehetnek szomszédjai u -nak, akik valaha értékelték i tartalmat. Ezek alapján jelölje $\mathcal{N}_i(u)$ u felhasználó i tartalomra vonatkoztatott legközelebbi szomszédjait. Ekkor r_{ui} értékelés az alábbi módon becsülhető [13]:

$$\hat{r}_{ui} = \frac{1}{|\mathcal{N}_i(u)|} \sum_{v \in \mathcal{N}_i(u)} r_{vi}$$

Azonban megfigyelve a fentebbi képletet észrevehető, hogy a szomszédok különböző hasonlóságainak figyelembe vételét nélkülözi. Így előfordulhat olyan eset, hogy a legközelebbi szomszéd olyan felhasználó lesz, akinél hasonlóság alapján van közelebbi. Ennek megoldására az alábbi, átdolgozott számítási módszer [13] használható:

$$\hat{r}_{ui} = \frac{\sum_{v \in \mathcal{N}_i(u)} w_{uv} r_{vi}}{\sum_{v \in \mathcal{N}_i(u)} |w_{uv}|}$$

Ahol a szomszédok már súlyozásra kerültek az alapján, hogy milyen hasonlósági tényezővel rendelkeznek u felé.

A **tartalom alapú** predikciós módszer u felhasználó i tartalomra adott r_{ui} értékelését becsüli meg felhasználva az i -hez leghasonlóbb tartalmakra adott értékeléseket. Az i -hez leghasonlóbb tartalmak a (tartalom-alapú) legközelebbi szomszédok [14]. Jelölje $\mathcal{N}_u(i)$ az u felhasználó által értékelt, i tartalomhoz leghasonlóbb tartalmakat. Az u által i -re adott r_{ui} értékelés becslése a következő [13]:

$$\hat{r}_{ui} = \frac{\sum_{j \in \mathcal{N}_u(i)} w_{ij} r_{uj}}{\sum_{j \in \mathcal{N}_u(i)} |w_{ij}|}$$

Mindkét módszer esetén szükség van normalizálásra, hogy azok ne vezethessenek az értelmezési tartományon kívüli eredményre. A hasonlósági mértékek a kapcsolódó irodalomban bevezetésre kerültekől választhatók ki a feladattól függően [14].

A dolgozatban alkalmazott kollaboratív szűrési módszer is felfogható egyfajta szomszédságalapú megközelítésnek, azonban a tervezett rendszerben egy gráf-alapú megközelítés kerül implementálásra.

3.2.3 A kollaboratív szűrés problémái

A kollaboratív szűrés problémái a következők [5]:

- Hidegindítási probléma (**új felhasználó problémája**): ha egy felhasználó még nem értékelt semmit a rendszerben (vagy az értékeléseinek száma rendkívül kevés), akkor a rendszer nem tudja effektíven megbecsülni a többi elem hasznosságát a felhasználóra tekintve, hiszen adatai nem összehasonlíthatóak a többi felhasználóéval.
- Hidegindítási probléma (**új tartalom problémája**): ha az adatbázisba új tartalom kerül, és nincs róla értékelési információ, akkor nem határozható meg a többi tartalommal kapcsolatos viszonya.
- Hidegindítási probléma (**értékelések hiánya**): a kollaboratív rendszerekben az értékelt tartalmak száma mindig elenyésző az összes tartalom számához képest. A kis tanítóhalmaz miatt pedig egyrészt rendkívül nehéz prediktálni a többi tartalom hasznosságát (nagy lesz a pontatlanság), másrészt nagyon sok tartalomra kell végrehajtani a predikciót. Ez a probléma a rendszer használatával csökken, de nem szűnik meg.
- **Egyediség problémája**: a kollaboratív rendszer a felhasználói értékelések alapján a felhasználók vagy a tartalmak hasonlóságából számít. Ha azonban egy-egy felhasználó vagy tartalom rendkívül eltérő ízlésű a többitől, a hasonlósági mérték nem megállapítható, vagy ha meg is állapítható, a pontatlanság nagyon nagy lesz.

3.3 Kontextus alapú ajánlás

A kollaboratív és a tartalom alapú ajánlással szemben a kontextus alapú ajánlás egy modernebb megközelítés: csak az utóbbi időkben tűnt fel és terjedt el. Létrejöttének és elterjedésének oka az, hogy a 2000-es évektől egyre több internetes közösségi

platform tűnt fel, melyek rendkívül nagy adathalmazt generálnak. Ennek az adathalmaznak a feldolgozása nem triviális, olyan nagy cégek köszönhetik az információfeldolgozási technikának a felemelkedésüket, mint a Google⁴, vagy a Netflix⁵.

A kontextus alapú ajánlás megpróbál a zenei objektumokra egy absztrakt objektumként tekinteni, és a környezetét leírni. Ez a környezeti adat sokféle lehet, például az értékelések ideje kapcsolódhat egy-egy eseményhez, mint egy koncert vagy film, mely felhasználta a zeneszámot. Formálisan: két elem hasonlóságát: $S(s_i, s_j)$ a két elem környezeti attribútumai határozzák meg. Ezt a hasonlóságot felhasználva pedig már meg lehet becsülni az s elem c felhasználóra vonatkoztatott $u(c, s)$ hasznosságfüggvényét.

Fontos megkülönböztetni a kontextus alapú ajánlást a tartalom alapú ajánlástól olyan módon, hogy míg a tartalom alapú ajánlás legtöbbször a digitális reprezentációból nyeri meta-adatait, addig a kontextus alapú ajánlás az internetes és online közösségi platformok által kínált információkra alapozva állapítja meg ezeket. Korábban a kontextus alapú rendszer környezeti attribútumait kézzel adták meg és keresték, azonban mára a világháló méretének és tartalmának köszönhetően készíthetők olyan adatbányászati algoritmusok, melyek egy-egy elemhez megtalálják a hozzá tartozó környezeti attribútumokat.

3.3.1 Zenei kontextusok által okozott nehézségek

A zenei ajánlórendszernél az ajánlási probléma megoldása nem feltétlenül elérhető csupán a predikciós probléma megoldásával, hiszen környezetileg hasonló zeneszámok keresésekor nagy valószínűséggel gyakoriak lesznek ugyanannak az előadónak a zeneszámai, illetve a közös albumról származó zeneszámok. A tartalom alapú ajánláshoz hasonlóan tehát itt is felmerül az előadó-hatás és az album-hatás

⁴ <http://www.google.com/>

⁵ <https://netflix.com/global/>

kiszűrésének feladata, mely szerint az ajánlási lista nem tartalmazhat egy azon előadótól, vagy egy azon zenei albumról származó zeneszámokat egymás után. Ennek megoldására a tartalom alapú ajánlás megoldásához hasonlóan a dolgozat elő- illetve utószűrés módszert javasol a (6.1) pontnak megfelelően.

3.3.2 A kontextus alapú ajánlás hátrányai

A kontextus alapú ajánlás attribútumai által generált hátrányok megegyeznek a tartalom alapú ajánlás hátrányaival, tehát az ajánlást nehezítik a morfológiai különbségek által okozott problémák, valamint a kevés információ és a túl nagy hasonlóság problémái.

4 Ajánlás közösségek részére

Az infokommunikációs ipar legnagyobb meghatározói manapság a közösségi hálózatok. Ezeken keresztül a felhasználók kapcsolatba léphetnek egymással, tartalmakat oszthatnak meg egymással. Ezen hálózatok információit a mai rendszerek csak részben használják ki, vagy teljesen mellőzik. Ezek a közösségi információk hatalmas információbázist jelentenek, melyek kihasználásával új lehetőségek és feladatok adódnak.

A közösségi információk kihasználásával kiszámítható az adott célközönség egy csoportjának aggregált, együttes döntése, melynek köszönhetően egy online vagy akár fizikai tartalomszolgáltató rendszer sokkal hatékonyabban szolgálhatja ki a közönségét.

4.1 A közösségi ajánlás, mint csoportos döntéshozási feladat

A közösségi információkon alapuló ajánlási módszernek mindenféleképpen meg kell határozni egyfajta csoportos véleményt. Ez a vélemény fogja képviselni minden, a csoportban lévő felhasználó értékrendjét, így olyan ajánlás adható, mely a hasznosságfüggvényt nem csak egy individuális felhasználóra, de az egész csoportra maximalizálja. Természetesen ez a maximum-érték többféleképpen értelmezhető: megadható például olyan heurisztika, mely alapján nem kerülhet be mellőzött zeneszám az ajánlási listába (hiába maximalizálná a közösségi értéket), de olyan is, ami tulajdonképpen egy egyszerű matematikai átlagolást végez.

4.1.1 Ismert, általánosan alkalmazható aggregációs módszerek

A központi feladat a csoportos ajánlás esetén az, hogy az individuális felhasználók által kinyilatkoztatott pozitív és negatív véleményt a teljes csoportra tekintve kell szemlélni. Erre a problémára nyújthatnak megoldást a klasszikus aggregációs módszerek [4], [15]:

- **Többségi szavazás:** a legtöbb (pozitív) szavazatot kapott elem kerül kiválasztásra. Ez tartalmakra alkalmazva negatív hatású lehet: kerülhet olyan

tartalom a lejátszási listába, melyet egy vagy több felhasználó korábban mellőzött.

- **Átlag:** klasszikus matematikai átlagot számol az összes értékelés között. Az előzőhöz hasonlóan bekerülhet olyan elem, melyet egy vagy több felhasználó mellőzött, ráadásul egy-egy kedvenc tartalom listába kerülésére is igen kevés az esély.
- **Borda-szám:** az elemek értékeléseiből pontszámokat generál az alábbi módszerrel: a középső értékhez a 0 párosul, majd a következő értékek 1-1 ponttal térnek el (a következő pozitív irányban 1 pontot kap, míg a következő negatív irányban -1-et).
- **Megerősítő szavazás:** megállapít egy küszöbértéket, és csak az afeletti értékeléssel rendelkező tartalmakat átlagolja, így megoldható, hogy a kevesebb szavazattal rendelkező tartalmak ne rontsák el a közösségi véleményt.
- **Legrosszabb eset:** a tartalmankénti legrosszabb értékelést veszi figyelembe. Ezzel a módszerrel könnyen elkerülhető, hogy a mellőzött tartalmak bekerüljenek az ajánlási listába, de arra nincs garancia, hogy az átlagosan legkedveltebbek képezik majd azt.
- **Az „egyenlőbb” felhasználó(k):** megállapít egy vagy több felhasználót, akinek a véleménye alkotja a csoportét, ő dönt mindenről.

Ahogy az látható, a módszerek legtöbbje alkalmazható, azonban mindegyiknek van hibája, és egyik módszer sem veszi igazán figyelembe azt, hogy a vélemények nem választhatóak el egymástól egy-egy küszöbértékkel (például a kedvenc és a kedvelt zeneszámok közötti határ nem éles, hanem folyamatos). A módszerek ettől függetlenül elő- és utószűrési feladatra kiválóak, hiszen segítségükkel elérhető az, hogy az ajánlási lista ne tartalmazzon mellőzött elemeket. Külön kiemelésre érdemes az utolsó, „egyenlőbb” felhasználó módszere, mely elsőre irrelevánsnak tűnhet, azonban ha megállapítható egy-egy csoportról, hogy a közösségi véleményük nagyban korrelál egy csoporttagéhoz, akkor rendkívül nagy számítási komplexitás

nyerhető azzal, ha a közösségi vélemény egyszerűen az adott individuális taghoz igazodik.

4.1.2 Az ismert megoldások korlátai, a közösségi ajánlás kihívásai

Média-tartalmak, főként zenei tartalmak esetén nem jellemezhető egy-egy felhasználó véleménye statikusan, hiszen ezek egy bizonyos pontatlansággal vannak jelen. A tetszés és a nem tetszés közti állapotátmenetet rendkívül nehéz jellemezni, hiszen ez az átmenet felosztható több állapotra: például kedvenc, kedvelt, átlagos, elfogadható, mellőzött stb. A változás emellett folyamatos, nem lehet megadni egy küszöbértéket, mely elvágja egymástól az állapotokat, mert jelen esetben egy folyamatos változás leírására van szükség. Ennek megoldására az állapotokat fuzzy-módszerrel kell leírni, a bemeneti állapotokat pedig tagsági függvényekkel ábrázolni. A számtani, súlyozott és egyéb átlagok számítása (aggregációs módszerek) hatásos lehet, de kimutatott [16], hogy ezek finom ábrázolása fuzzy-módszerrel sokkal hatásosabb.

4.2 Az állapotátmeneteken alapuló vélemények aggregálása

Korábbi munkákban megállapításra került a vélemények közti hasonlóság és távolság definíciója [17] [18]. Ezek a definíciók elsőre igen triviálisnak tűnhetnek (mint például az euklideszi- vagy a Manhattan-távolság), azonban nagyon fontos kihangsúlyozni azt a tényt, hogy itt változó állapotátmeneteket leíró változókról van szó, melyek között nem húzódik éles határ: az átmenet folyamatos (fuzzy). Ezen változók megfelelő aggregációját a távolságuk és a hasonlóságuk mértéke adja [16].

A közösségi ajánlás, mint döntéshozási probléma megoldásaként tehát meg kell határozni a felhasználók véleményét reprezentáló fuzzy-értékeket, a köztük lévő hasonlóság és távolság megfelelő értékét, majd ezeket felhasználva megadható a legjobb döntés.

4.2.1 Hasonlóság közösségi vélemények között

A korábbi munkák által meghatározott hasonlósági mérték közösségi vélemények között a következőképpen írható fel [17]:

$$S(\hat{R}_i, \hat{R}_j) = \frac{\int_x \left(\min \{ \mu_{\hat{R}_i}(x), \mu_{\hat{R}_j}(x) \} \right) dx}{\int_x \left(\max \{ \mu_{\hat{R}_i}(x), \mu_{\hat{R}_j}(x) \} \right) dx}$$

ahol:

- $\mu(x): R \rightarrow [0,1]$ a tagságfüggvény,
- $\hat{R}_i = (a_i, b_i, c_i, d_i)$ pozitív trapezoid fuzzy szám ($a_i < b_i < c_i < d_i$), az i -edik tudós véleménye

Ekkor azonban előfordulhat, hogy a hasonlósági mérték olyan esetben is megegyezik, amikor annak nem lenne szabad. Ennek elkerülése végett bevezetésre kerül minden elem súlya $x \in R$, melynek köszönhetően $\hat{R}_i, \hat{R}_j$ fuzzy-számokra, $0 \leq w(x) \leq 1$ esetén $\hat{R}_i, \hat{R}_j$ közti hasonlósági mérték a következőképpen írható fel [16]:

$$S(\hat{R}_i, \hat{R}_j) = \frac{\int_x \left(w_{\min}(x) \min \{ \mu_{\hat{R}_i}(x), \mu_{\hat{R}_j}(x) \} \right) dx}{\int_x \left(w_{\max}(x) \max \{ \mu_{\hat{R}_i}(x), \mu_{\hat{R}_j}(x) \} \right) dx}$$

ahol $w_{\min}(x), w_{\max}(x)$ jelöli az adott (konzisztens) véleménytér súlyfüggvényét.

Mivel azonban $\min \{ \mu_{\hat{R}_i}(x), \mu_{\hat{R}_j}(x) \}$ és $\max \{ \mu_{\hat{R}_i}(x), \mu_{\hat{R}_j}(x) \}$ tagsági függvényei a véleménytérnek, és $w_{\min}(x) = \min \{ \mu_{\hat{R}_i}(x), \mu_{\hat{R}_j}(x) \}$, $w_{\max}(x) = \max \{ \mu_{\hat{R}_i}(x), \mu_{\hat{R}_j}(x) \}$, így a képlet a következőképpen módosul [16]:

$$S(\hat{R}_i, \hat{R}_j) = \frac{\int_x \left(\min \{ \mu_{\hat{R}_i}(x), \mu_{\hat{R}_j}(x) \} \right)^2 dx}{\int_x \left(\max \{ \mu_{\hat{R}_i}(x), \mu_{\hat{R}_j}(x) \} \right)^2 dx}$$

A módosított egyenlettel az eredmény sokkal pontosabb és kielégítőbb, több esetben ad helyes megoldást.

4.2.2 Távolság közösségi vélemények között

A vélemények közti távolság rendkívül fontos eleme az aggregációs eljárásnak. A távolság kiszámítására több módszer is alkalmazható, például a kapcsolódó irodalomban kifejtett távolságfüggvények [14], vagy a speciális fuzzy-távolságként bevezetett távolság [19]. A dolgozat a közösségi vélemények közti távolságra a kapcsolódó irodalom által bevezetett speciális, normalizált Hamming-távolságot javasolja [16]:

$$d(\hat{R}_i, \hat{R}_j) = \frac{D(\hat{R}_i, \hat{R}_j)}{D^*(\hat{R}_p, \hat{R}_q)}$$

ahol $\hat{R}_i = (a_i, b_i, c_i, d_i)$, $\hat{R}_j = (a_j, b_j, c_j, d_j)$ a fuzzy-változók reprezentálják a véleményeket, melyekből kiszámítódik minden pár, majd ezekből $D^*(\hat{R}_p, \hat{R}_q) = \max D(\hat{R}_i, \hat{R}_j)$ adja a legnagyobb távolságot, így a fenti képlet tulajdonképpen egy normalizált Hamming-távolság, mely eleget tesz a Hamming-távolsággal szemben megfogalmazott követelményeknek [13]:

- $0 \leq d(\hat{R}_i, \hat{R}_j) \leq 1$
- $d(\hat{R}_i, \hat{R}_j) = 0 \leftrightarrow \hat{R}_i = \hat{R}_j$
- $d(\hat{R}_i, \hat{R}_j) = d(\hat{R}_j, \hat{R}_i)$
- $d(\hat{R}_i, \hat{R}_j) \leq d(\hat{R}_i, \hat{R}_k)$ és $d(\hat{R}_j, \hat{R}_k) \leq d(\hat{R}_i, \hat{R}_k)$, ha $\hat{R}_i \subseteq \hat{R}_j \subseteq \hat{R}_k$

Ez a fajta számítási módszer nem olyan komplex, mint a kapcsolódó irodalomban találhatóak, azonban az ajánlás megadásához megfelelően pontos, és számítási igénye jóval kisebb.

4.2.3 A vélemények aggregációja a hasonlóság és a távolság segítségével

Az aggregáció megadja a közösség véleményéhez legjobban illeszkedő, összegzett véleményt. Ez nagyon jól alkalmazható egy szórakozóhely vendégei igényeinek kielégítésére, vagy akár sokmillió felhasználószámú tartalomszolgáltató rendszerek esetén is, hiszen az összegzett és megfelelően szűrt értékelésekre és tetszési mutatókra építve a hasonlóságok és távolságok kiszámítása után meghatározható, hogy az adott közönségnek mi a legmegfelelőbb tartalom.

A kapcsolódó irodalomban megállapított aggregáció a következő [17]:

$$\hat{R} = F(\hat{R}_1, \hat{R}_2, \dots, \hat{R}_n) = \sum_{i=1}^n w(E_i) \odot \hat{R}_i$$

ahol $\odot$ a fuzzy-szorzás operátor.

A fenti képlet megadja a vélemények egyenrangú aggregációját, azonban gyakorlatban megfigyelhető, hogy a csoportos döntéshozás eredménye, és ezzel pontossága nagymértékben függ a résztvevők szerepének (értékeléseinek) mértékétől. Általánosan érdemes tehát megtalálni azokat a résztvevőket, akik tapasztaltabbak, pontosabb és jobb döntést hoznak (a közösségre nézve), mint a csoport többi tagja. Ezeket a résztvevőket *hozzáértőként* kezeli a rendszer. Általánosan megadható tehát minden résztvevőhöz egy súly, mely nemcsak a hozzáértését, hanem a döntésbeli beleszólásának mértékét is reprezentálja. Ezek a súlyok $s \in [0,1]$ közötti valós számok, melyek összege 1-et ad:

$$\sum_{i=1}^n s_i = 1$$

Ennek következtében a fent bevezetett konzisztencia mértéket, és a dolgozat által bevezetett súlyok a következő képletben szintetizálódnak:

$$\hat{R} = F(\hat{R}_1, \hat{R}_2, \dots, \hat{R}_n) = \sum_{i=1}^n s_i w(E_i) \odot \hat{R}_i$$

ahol $\odot$ a fuzzy-szorzás operátor.

4.3 Közösségek, mint a szolgáltatóipar valós közösségei

Ahogy az már a célkitűzések között is szerepelt, a dolgozat célja a nagy felhasználószámú tartalomszolgáltató rendszerek, illetve a szórakozóhelyek közösségeinek minél jobb kiszolgálása. Tartalomszolgáltató rendszer esetén ez a csoport jelenthet adott térségben tartózkodó felhasználókat, esetleg adott demográfiai sávba tartozó embereket, szórakozóhely esetén pedig az éppen aktuális közösséget. Ezeknek köszönhetően a tartalomszolgáltató rendszereknek és a szórakozóhelyeknek

történő csoportos döntéshozás feladata nagyban különbözik a klasszikus csoportos döntéshozástól, így új feladatokra kell megoldást találni.

4.3.1 Korábbi döntések szerepe az ajánlásban

A szolgáltatóiparban, főként az ajánlórendszerek által kiszámított csoportos döntéshozás sokban különbözik a klasszikus közösségi döntés feladatától. Az általános döntéshozáskor az adott állapot bemenetei nem függenek az előző állapot



kimenetétől, tehát a rendszerben nem található visszacsatolás.

7. ábra: A klasszikus csoportos döntéshozás folyamata

felhasználók a következő lista előtt értékelik, így elmondható, hogy a közösségi ajánlást végző rendszer bemenete függ az előző állapottól.



8. ábra: Csoportos döntéshozás folyamata a szolgáltatóiparban

Példaként *klasszikus csoportos döntéshozás* egy konferencia időpontja. A konferencián részt vesznek adott emberek, akiknek vannak kizárt időpontjaik (ekkor elfoglaltak), valamint több szintben preferált más időpontjaik (az időpontokra lehet szavazni). A szavazások megtétele után a rendszer kiszámolja a teljes csoportnak legalkalmasabb időpontot, majd megrendezésre kerül a konferencia. *Tartalmakat szolgáltató piaci döntéshozás* esetén ezzel szemben egy szórakozóhely esetén a résztvevő felhasználók szavaznak (a zeneszámokra lehet szavazni), majd kigenerálódik az ajánlási lista. A listára azonban újra szavazhatnak, mely befolyásolja a következő csoportos döntést.

4.3.2 Dinamikus közönség

A csoportos döntéshozás közönsége is sokban különbözik a klasszikus megoldások esetén, hiszen általánosan a felhasználók nem változnak, a csoport statikus (legfeljebb új taggal bővíthet), egy szórakozóhely vagy tartalomszolgáltató rendszer esetén viszont az aktuális csoport tagjai folyamatosan változnak - óránként akár a teljes közönség is - ezáltal kialakulhat egy teljesen más ízlésű csoport, akiknek az előző ajánlás nem felel meg. Számon kell tehát tartani a csoport változását is, és bizonyos mértékű változás felett újraszámolni a teljes ajánlást, hogy az új tagok igényeit elégítse ki, és ne tanuljon „félre” az algoritmus.

A közönség még egy szempontból nagyban különbözik: míg a klasszikus csoportos döntéshozás esetén minden résztvevő szavaz, tehát a teljes közönség ismert, addig az nem várható el, hogy egy szórakozóhely teljes közönsége használja a szavazásra lehetőséget adó alkalmazást. Ezen a ponton bukik el a legtöbb hasonló rendszer: ha egy több száz, vagy több ezer fős rendezvényen csak néhány ember használja az alkalmazást, akkor a teljes valós közönség ízlését ők képviselik. Ez megengedhetetlen, így számon kell tartani, hogy a közönség hány százaléka használja az alkalmazást. A jelenleg piacon lévő alkalmazások másik nagy hibája, hogy túl nagy szabadságot kap a közönség, ezáltal a szórakozóhely alapvető irányultsága eltűnhet: biztosítani kell a szórakozóhely számára a lista módosítását oly módon, hogy az alapvető preferenciája, alap-profilja nagyban befolyásolja azt, ne lehessen eltéríteni.

4.3.3 Vélemények súlya

A résztvevők egy csoportos döntéshozásban alapvetően egyenlőek. Nagyon fontos azonban, hogy a szórakozóhelyek, vagy bármely hasonló szolgáltatók rendelkeznek egy alap arculattal, egyfajta profillal, melyet nem szeretnének megváltoztatni, hiszen nem szeretnék elveszteni régóta meglévő törzsközönségüket. Ennek következtében a tartalmakra irányuló döntéshozásban a csoportot összefogó egységnek sokkal nagyobb hangsúlyú véleményt kell biztosítani, mint az aktuális közönségnek. Ezen kívül a bizalom fontosságát figyelembe véve biztosítani kell azt, hogy az adott egység

(például szórakozóhely) módosíthassa az ajánlási listát: így biztosított az, hogy nem kerül lejátszásra olyan elem, amit nem szeretne.

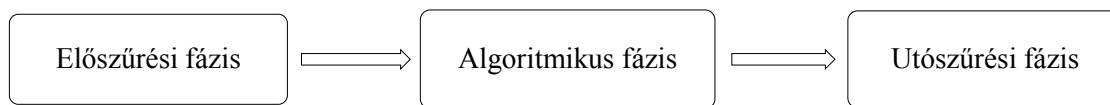
A szervezeti egység mellett fontos kitérni a törzsközönségre is, hiszen az ő ízlésük a legfontosabb minden szolgáltató egység számára. A törzsközönség véleménye úgy reprezentálható a legjobban, hogy a minél régebb óta odajáró, és minél több időt ott töltő felhasználók véleménye jobban számít, mint az újaké.

5 Algoritmikus módszerek, alkalmazott metrikák

Az alábbi fejezet bemutatja a kifejlesztett megoldás algoritmusait, beleértve az alkalmazott metrikákat. Nagyon fontos megjegyezni, hogy a fejezetben leírt, negatívumként feltüntetett korlátok és megoldási hibák a jelenleg használt, nyilvános megoldásokra igazak, a leírt megoldás részleteiben alkalmazható lenne akár más reprezentációval is, mely javíthatna egy meglévő módszeren, azonban a cél egy prototipikus modell megvalósítása. A fejezet által bemutatott módszerek kitérnek mind az individuális, mind a közösségi ajánlásra.

5.1 Az ajánlási folyamat áttekintése

Az ajánlási folyamat három fő lépésből áll: egy *előszűrésből*, egy *algoritmikus mértékszámításból*, és egy *utószűrésből*. Ezen lépések megkülönböztetése kiemelten fontos, hiszen a modern tartalomszolgáltató rendszerben gyakran vannak megkülönböztetett tartalmak (melyet például egy rendkívül fontos ügyfél tölt fel), melyek helyét a listában biztosítani kell. Emellett felkerülhetnek olyan tartalmak, melyek megtekintése/meghallgatása bizonyos országok, életkorszerű csoportok stb. számára nem megengedett, így ezeket szűrni kell. A folyamatot az alábbi ábra mutatja be:



9. ábra: Az ajánlási folyamat fő fázisai

5.1.1 Előszűrés

Az előszűrés több szempontból is fontos fázis a teljes ajánlási folyamatban:

- **Kiszűri** azokat a tartalmakat, melyeket a felhasználó **nem fogyaszt** jogi, társadalmi vagy más egyéb okokból. Ilyen ok lehet például, hogy egy adott országban tiltott az adott tartalom.
- Lehetőséget az algoritmikus fázis (2.) segítésére az **album-hatás és az előadó-hatás** problémájának megoldásával: kiszűrésre kerülhetnek azok az előadók/albumok, akiknek műveit az aktuális (vagy egy nem túl régi) ajánlási lista tartalmazza. Ez egyrészt komplexitást csökkent, másrészt **segíti az ajánlási lista változatosságát**.
- Elvégzi a felhasználók (beleértve az aktuális tartalomszolgáltató – például szórakozóhely) **véleményeinek súlyozását**, még mielőtt az algoritmikus fázis (2.) számolna ezekkel.
- **Normalizálja** a vélemények értékeit, így egyenlően számíthatnak azok a felhasználók, akik már régóta használják a rendszert, és azok, akik még csak most ismerkednek vele (fontos megjegyzés: attól még a törzstagok súlya jobban számít, ez a szolgáltató egységtől független, a rendszer használatára általánosan vonatkozik).

Az előszűrési fázisnak van még egy nagyon fontos funkciója: a tartalomszolgáltató rendszereknél nagyon gyakran van „professzionális” előfizetés, melyre az adott cég nagyobb figyelmet fordít. Esetleges kérések esetén a fizetett tagok extra igényei könnyen biztosíthatóak az előszűrési fázisban.

5.1.2 Algoritmikus fázis

Az algoritmikus fázis végzi az ajánlási feladatok megoldását. Kiszámolja a szükséges értékeket, majd előállítja a *kibővített ajánlási listát*. Az egyes kiszámítandó értékek a következők:

- **Felhasználók hasonlóságának megállapítása:** ebben a fázisban megállapításra kerül egy-egy metrika a felhasználók között. Ezen metrikák azt a célt szolgálják, hogy egy-egy felhasználóhoz meg lehessen állapítani a leghasonlóbbakat, mely fontos szerepet játszhat a vélemények kiértékelésénél.
- **Tartalmak hasonlóságának megállapítása:** hasonlóan a felhasználók hasonlóságához, ez a fázis megad egy metrikát a tartalmak (jelen esetben zeneszámok) között az azokra adott értékelések, valamint az azokat jellemző attribútumok (meta-adatok) alapján.
- **Predikció végrehajtása:** a központi ajánlási feladat megoldása. Megállapításra kerül a kibővített ajánlási lista (mely a következő fázisban kerül leszűrésre) az előző metrikák, és az előző listára adott értékelések alapján. A predikciós folyamat természetesen az alapvető céloknak megfelelően felbontható individuális felhasználóknak történő ajánlásra is szórakozóhelyek valós közönségeinek kiszolgálására is.

5.1.3 Utószűrési fázis

Az utószűrési fázis az ajánlási lista teljes személyre szabását, finomhangolását hajtja végre.

- Megállapítja, hogy a felhasználónak mik a tetszési indexei a **különböző típusú tartalmakhoz**, majd a tartalmak típusainak **arányát** úgy módosítja a listában, hogy az a felhasználónak a legmegfelelőbb legyen.
- Elvégzi a **tartalom-alapú finomhangolást**, melynek célja, hogy az ajánlási listában található tartalmak megfelelő sorrendben helyezkedjenek el. Ilyen finomhangolás lehet például a zeneszámok hangnemeinek összepárosítása (vagy más, rendelkezésre álló információk párosítása)

- **Normalizálja** a tartalmakat, például zeneszámok esetén a zeneszámok hangerőértékeit, hogy azok megfelelően illeszkedjenek egymáshoz.

5.2 Rendelkezésre álló adatok, ajánlási attribútumok

Ahogy az korábban jól látható vált, a dolgozat keretein belül fejlesztett ajánlórendszer a teljes megoldás egy alrendszere, mely saját feladatait látja el, a megfelelő adatokat jól definiált interfészen keresztül éri el és biztosítja a központi – repository – egységnek. Az ajánlást végző alrendszer számára nélkülözhetetlen adatok a felhasználók azonosítói, a tartalmak azonosítói, a felhasználókat és a tartalmakat leíró elsődleges attribútumok (meta-adatok), valamint a felhasználók preferenciái (értékelései) a tartalmakkal szemben. Az adatok a következők:

- **Felhasználói és zenei azonosítók:** a felhasználói azonosítók felelnek a felhasználók, előadók és szórakozóhelyek azonosításáért. A felhasználó típusát az adott azonosító értéktartománya adja meg, például a negatív értékek a szórakozóhelyeket jelölik. A zenei azonosítók reprezentálják a zeneszámokat, minden zeneszámhoz egyedi azonosító társul.
- **Felhasználói attribútumok:** a felhasználóról számos adat kerül nyilvántartásra, azonban ezeknek csak kis része befolyásolja az ajánlást. Az adatok közé tartoznak demográfiai adatok és a személyes adatok is.
- **Közösségi attribútumok:** a közösségek felhasználók csoportjai. A közösségi attribútumok megegyeznek a felhasználói attribútumokkal, tulajdonképpen a felhasználói azonosítókat reprezentálják, melyeken keresztül elérhetőek a felhasználókról rendelkezésre álló információk.
- **Zenei attribútumok:** a zeneszámokat jellemző adatok a meta-adatok, valamint a zeneszám nélkülözhetetlen jellemzői, mint például a címe vagy az előadója (a zeneszám címe és előadója nem lehet meta-adat). A meta-adatok nagyon sokrétűek, általában megtalálható köztük egy-vagy több zenei stílus, de akár a felvétel helye, tempója, ideje, vagy éppen a dalszöveg nyelve.
- **Értékelések:** a felhasználók és a zeneszám interakcióját jelző adatok a felhasználók zeneszámra adott értékelései. Nagyon fontos, hogy nem csupán

az értékelések, hanem azok sorrendje is számít, így megfelelő sorrendben kerülnek átadásra az ajánlást végző alrendszernek. Az értékelés lehet pozitív, negatív, vagy nulla, azonban – későbbi indoklást alapul véve – a felhasználó – zeneszám kapcsolat nem egy statikus értéként, hanem állapotátmenetként, fuzzy-ként kerül leírásra.

5.3 Reprezentáció, az alkalmazott tárolási módszer

Az ajánlási algoritmusok a rendelkezésre álló adatokat, mint bemeneteket egy adatstruktúrába szervezik. Ezen adatstruktúrán kerülnek elvégzésre azok a matematikai műveletek és transzformációk, melyek egymás utáni alkalmazásával megkapható eredményként a kívánt ajánlási lista, vagy valamely egyéb – ajánlási – feladat megoldása. Ezek az adatstruktúrák általában ritkás mátrixokként kerülnek megvalósításra, azonban megadhatóak gráfként, esetleg hipergráfként is.

Mielőtt azonban leírásra kerülnének az általánosan alkalmazott reprezentációs módszerek, nagyon fontos megemlíteni a már korábban (5.1.2) is megállapított tényeket, mint például, hogy egy-egy felhasználó egy-egy tartalomhoz tartozó preferenciaértéke nem feltétlenül egy adott statikus érték, hiszen a különböző állapotok közötti átmenetek nem élesek, hanem folytonosak. Az állapotok között tehát nem lehet egyértelműen olyan küszöbértékeket definiálni, mely elválasztja a felhasználónak kiemelten tetsző és „egyszerűen” tetsző tartalmakat, ezek között átmeneteket kell definiálni.

A kidolgozott modell kétértékű értékelést támogat. Ennek megfelelően a felhasználó pozitívan, vagy negatívan tudja értékelni az adott zeneművet, de a fentiekben megfogalmazott okok miatt ez a két (vagy több, esetleg 5) érték nem feltétlenül a legmegfelelőbben fejezi ki az adott véleményt. Ennek a feladatnak a megoldása kétféleképpen zajlik:

- Értékeléskor a vélemények nem csupán az adott zeneszámmra vannak hatással, hanem a zeneszámokhoz kapcsolódó egyéb adatokra is, mint például a hasonló zeneszámok, címkék (a jellemző meta-adatok), esetlegesen a kapcsolódó felhasználók.
- A vélemények kiskálázása: a vélemények a $[0,1]$ zárt intervallumból átskálázódnak a $[0,1000]$ intervallumra. Ez értelemszerűen nem egy értékelés eredménye, hanem az értékelések összhathása. Ennek köszönhetően lehetőség van az állapotátmenetek sokkal finomabb ábrázolására, mely segítségével sokkal jobban jellemezhető az adott felhasználó.

5.3.1 Mátrix alapú reprezentáció, a mátrix-reprezentáció hátránya

A mátrix alapú reprezentáció a klasszikus ajánlási algoritmusok által leggyakrabban alkalmazott módszer. A bemeneteket (felhasználói azonosítók, tartalomazonosítók, értékelések) olyan módon szervezik mátrixba, hogy a sor-oszlop azonosítók értéke a felhasználó és a tartalom azonosítója legyen, melynek megadásával lekérdezhető az adott preferenciaérték. A mátrix természetesen rengeteg 0 elemet tartalmaz, melyek a még nem értékelt tartalmak (ha a 0 aktuálisan egy értékelési lehetőség, akkor *null*, vagy esetleg más egész szám reprezentálja ezt az értéket).

11	12		14				
	22	23		25			
31		33	34				
	42			45	46		
				55			
				65	66	67	
				75		77	78
						87	88

10. ábra: Az értékelések reprezentálása ritkás (sparse) mátrixként

A ritkás mátrixok rendkívül hatékony megoldások az adatok tárolására, azonban az ezekre alapuló rendszerek – mint például az Apache Mahout – egy rendkívül nagy, a későbbiekben akár végzetes problémával küzdenek. Szükséges letárolni minden valaha megtett értékelést, melyek az idő folyamán lineárisan nőnek, így – főleg egy

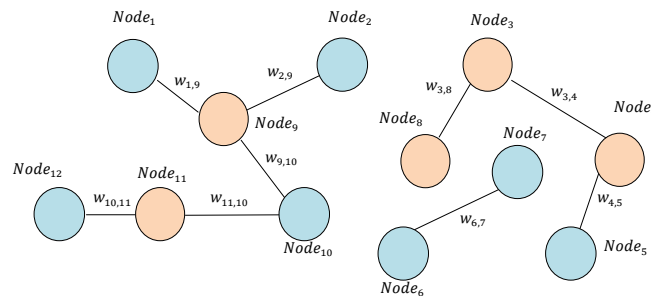
nagy tartalomszolgáltató rendszer esetén elkerülhetetlen a sok százmillió, vagy akár milliárdos nagyságrendű sor letárolása, mely súlyos problémákhoz vezet.

A mai nagy tartalomszolgáltató, illetve közösségi rendszerek (például Facebook, Twitter) kénytelenek voltak átállni fa-elrendezésű, NO-SQL adatbázisokra, és bizonyítottan a mai napig legnagyobb problémájuk a fentebb felvázolt feladat megoldása.

Bár adható olyan, gráfot reprezentáló mátrix-felírás, mely a problémát csökkenti, sőt, esetlegesen meg is oldja, a klasszikus, főként a kollaboratív szűrés hatására elterjedt reprezentációs módszer erre nem képes.

5.3.2 Gráf alapú megközelítés:

Az egyik jól alkalmazható reprezentációs módszer egy gráf, mely lehet irányítatlan vagy akár irányított is, de mindenképpen súlyozott. A gráf csomópontjai reprezentálják a megfelelő adatokat (általában felhasználókat és a tartalmakat), az élek pedig a köztük lévő súlyt, azaz a felhasználó értékeléseit az adott tartalmakra.



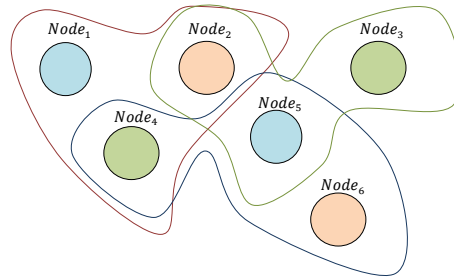
11. ábra: Gráf alapú reprezentáció: a színek a különböző típusú csomópontokat jelölik

A gráf alapú reprezentáció egyre népszerűbbé válik könnyű áttekinthetősége miatt, azonban alapjaiban véve – ha figyelmesen megnézzük – nem különbözik a ritkás mátrixtól, csupán egy másfajta megvalósítása annak. Nagyon fontos azonban megjegyezni, hogy a gráfokra alkalmazhatóak a már kidolgozott gráf bejárasi – és egyéb algoritmusok, melyek megvalósítását megkönnyíti a reprezentáció. A dolgozat

a későbbiekben bemutat egy olyan megoldást, mely az értékelések tárolásakor nem lineárisan, hanem csupán logaritmikusan növeli a letárolandó sorokat, mely a jelenlegi – nyilvános – algoritmusokkal szemben így alkalmazhatóbb lesz a megfelelő célokra.

5.3.3 Hipergráf alapú reprezentáció

A hipergráfok olyan gráfok, melyekben az élek nem csupán kettő, hanem $k \in N, k \geq 2$ csomópontot kötnek össze. A kapcsolódó irodalomban megfigyelhető néhány új keletű, hipergráfokat használó reprezentációs módszer is [20]: ahelyett, hogy az adott gráfban az élek egy-egy felhasználó-tartalom párost kötnének össze a preferenciaértékkel, mint az él súlya, a hipergráf alapú megoldásokban egy-egy él több csomópontot is tartalmazhat, és ezekhez közösen rendel egy megfelelő súlyt.



12. ábra: Hipergráf alapú reprezentáció: a különböző színű csomópontok a különböző típusú tartalmakat jelölik

Megfigyelhető, hogy a lehetőségek mértéke így megsokszorozódik: amit egy-egy korábbi megvalósításban lehetetlen volt megkülönböztetni, az hipergráfokkal már teljesen egyértelmű. Jó példa a két felhasználó, két meta-adat és két zeneszám típusú csomópont megfelelő kapcsolata. Míg az első megoldásban a hagyományos gráf 6 csomópontot és 6 élt tartalmaz 2 felhasználó, 2 meta-adat és 2 zeneszám esetén úgy, hogy az értékelések nem megkülönböztethetők, addig a hipergráf alapú megoldás 6 csomóponttal és 3 éllel egy olyan modellt ad, melyben az értékelések egyértelműen megkülönböztethetők.

Bár a hipergráf alkalmazása egy rendkívül érdekes megoldás, sajnálatos módon az implementálandó algoritmusok komplexitása megnő, így alkalmazása nagy felhasználószámú tartalomszolgáltató rendszerek esetén kérdéses.

5.4 Használt reprezentáció, alkalmazott metrikák

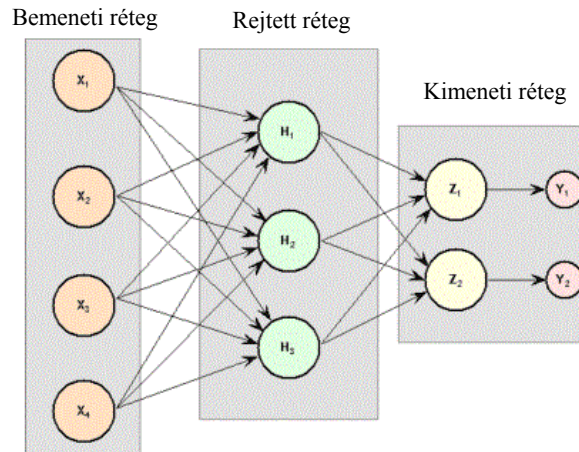
A tervezett rendszerben a felhasználó és zenei azonosítók, a megfelelő jellemző attribútumok (meta-adatok), és az értékelések egy hagyományos gráfban kerülnek letárolásra. A gráf reprezentálja az összes típusú csomópont közti kapcsolatot, így a 3 típusú csomóponthoz 6 féle kapcsolat állapítható meg:

- **Felhasználó – felhasználó típusú kapcsolat:** a felhasználó-felhasználó típusú kapcsolat reprezentálja a felhasználók közti hasonlóságot. Az adott kapcsolat súlya jelképezi azt, hogy a felhasználók ízlése mennyire hasonlít. Az adott metrika tehát a felhasználókat jellemző attribútumok (meta-adatok) metrikáinak megfelelő aggregációja (tulajdonképpen matematikai átlaga), melyhez a később definiált meta-adat – meta-adat közti távolság kerül felhasználásra.
- **Felhasználó – tartalom típusú kapcsolat:** a felhasználó – tartalom típusú kapcsolat reprezentálja a már megejtett értékeléseket, valamint a prediktált preferenciaértékeket. Ezen értékek alapján adható egyértelmű személyre szabott ajánlás, ezek reprezentálják az alapvető ajánlási feladat megoldását. Az adott értékelés kétértékű lehet: tetszik/nem tetszik $(-1,1)$, továbbá a profilgeneráláskor minimum-maximum $(\min, \max)$. Amennyiben egy felhasználó újra értékeli egy adott zeneszámot, a két értékelés összeadódik, így a kapható értékek száma növekszik $(-2,-1,0,1,2)$.
- **Felhasználó – meta-adat típusú kapcsolat:** a kapcsolat reprezentálja egyrészt a felhasználó jellemzőit, másrészt a felhasználó által előnyben részesített zenei jellemzőket. A felhasználó pozitív értékelései által közelebb kerülnek hozzá az adott zeneszámot reprezentáló meta-adatok, így az ajánlás pontossága nagyban növelhető. A felhasználó – meta-adat kapcsolat akkor

kerül bekötésre, ha a felhasználót jellemzi az adott adat. Ez a kapcsolat nagy segítség lehet a felhasználók közti hasonlóság kiszámításában.

- **Tartalom – tartalom típusú kapcsolat:** ez a kapcsolat szimbolizálja a zeneszámok hasonlósági mértékét. Minél nagyobb szám szerepel a súlyon, annál hasonlóbb két zeneszám. A zeneszámok hasonlóságát egyrészt a megfelelő felhasználó által rájuk adott értékelések hasonlósága adja, másrészt az azokat jellemző meta-adatok hasonlósága. Amennyiben egy adott felhasználó pozitívan értékeli „A” zeneszámot, majd „B” zeneszámot, akkor a két zeneszám közti hasonlósági mérték nő, amennyiben azonban „A” zeneszámot pozitívan, még „B” zeneszámot negatívan értékeli, a különbség nő. Az értékelési metrika egy tulajdonképpeni és-kapunak feleltethető meg.
- **Tartalom – meta-adat típusú kapcsolat:** az adott zeneszámot jellemző meta-adatokat adja meg. A súly akkor kerül bekötésre, ha az adott zeneszám rendelkezik az adott meta-adattal. A meta-adat a felhasználó – meta-adat kapcsolathoz hasonlóan itt is jellemezhet akár felhasználót is, ekkor a súly azt reprezentálja, hogy az adott meta-adattal rendelkező felhasználók mennyire kedvelik az adott tartalmat általánosan.
- **Meta-adat – meta-adat típusú kapcsolat:** a meta-adatok közti hasonlóságot megadó kapcsolat. Nagy szerepet játszik a zeneszámok és a felhasználók közti hasonlóság megállapításában. A meta-adat – meta-adat kapcsolatok az adatbázisban inicializálva szerepelnek (a MusicBrainz⁶ eleve tartalmazta ezeket), az újonnan bekerülő meta-adatok kapcsolatainak súlya pedig egy előrecsatolt neurális hálózat alkalmazásával kerülhet megállapításra, azonban az alkalmazott hardverkörnyezet limitáltsága miatt ennek megvalósítására, tesztelésére és egyúttal mélyebb kidolgozására nem volt lehetőség.

⁶ <http://musicbrainz.org>



13. ábra: Előrecsatolt neurális hálózat

Az ábrán látható neurális hálózat kezdeti súlyai a MusicBrainz adatbázisból származó meta-adat hasonlóságok. Amennyiben egy új meta-adat kerül a hálózatba, úgy a neurális hálózatnak köszönhetően megállapítható, hogy miként viszonyul a többi attribútumhoz.

Egy előrecsatolt neurális hálózat alkalmazható lenne akár az ajánlás elvégzésére, azonban ekkora adatmennyiséggel ennek alkalmazása fizikailag nem megvalósítható. Ezen kívül fontos megjegyezni, hogy az előre csatolt neurális hálózat csak későbbi fejlesztésként kerülhet a rendszerbe annak nagymértékű erőforrásigénye miatt.

A mátrixreprezentációnál már említett tárolási feladat úgy kerül megoldásra, hogy a felhasználó által adott értékelés hatására a gráfban lévő két csomópont között létrejön, vagy változik az adott súly mértéke. Miután a gráfban letárolásra került, az érték törölődhet a klasszikus $\langle c, s, v \rangle$ hármast reprezentáló adattárolóból (rendszerint adatbázistáblából). Ezzel már csökkent a komplexitás, ehhez azonban még hozzájön a későbbiekben tárgyalt nélkülözhető értékek levágása, melynek működése hasonló az infokommunikációban alkalmazott kódolásokéhoz.

5.4.1 Klasszikus és gráf-alapú hasonlósági metrikák

Az ajánlórendszerek algoritmusai által felhasznált hasonlósági mértékek a pontosság egyik legfőbb meghatározói. Ezek a metrikák általában a Pearson-korreláció, Manhattan-távolság, vagy valami más, hasonló módszer. Ezek a módszerek rendkívül elterjedtek pontosságuk miatt, azonban egy tartalomszolgáltató rendszer esetén több szempontból is megkérdőjelezhető a használatuk, így a tervezett rendszer a fentebb bevezetett kapcsolatok súlyaira támaszkodik.

Az első szempont az, hogy a tartalomszolgáltató rendszerekben rengeteg (sokmillió) tartalom áll rendelkezésre, melyek ajánlásához szükséges megállapítani a bemenetként szolgáló tartalom és az összes többi tartalom hasonlósági mértékét, melyből egy maximumválasztási módszerrel (mely lehet akár legközelebbi-szomszéd módszer is, de alkalmazható egyszerű maximumkeresés is) kikerül a kért számú tartalom azonosítója. Ez egyrészt rendkívül időigényes folyamat ahhoz, hogy online végre lehessen hajtani, másrészt a hasznosságfüggvény matematikai maximalizálása még nem feltétlenül elég a felhasználói elégedettség maximalizálásához.

A másik fő indok, hogy a tartalomszolgáltató rendszereknek nem elég csupán kiszámítani az ajánlott tartalmakat, át is kell adni azok azonosítóit, le kell kérdezni, hogy mit azonosítanak, valamint meg kell jeleníteni az alkalmazási rétegnek. Ez egy időigényes folyamat, így az ajánlási időnek minimálisnak kell lennie: ennek megoldására a hasonlóság gyakran offline számolódik, majd egy megfelelő tárolóból (általában adatbázistábla) kerül lekérésre az eredmény. Azonban egy ilyen tárolás esetén a jelenlegi megoldási módszerek kiszámítják minden elempár hasonlósági mértékét, melyet letárolnak – jobb esetben csak a $k \in N$ legnagyobb, azonban sokszor előfordul, hogy mindet. Ez ötmillió tartalom esetén, a legkisebbnek számító $\langle integer, integer, byte \rangle$ tárolással még az adott programnyelvre jellemző overhead nélkül is kerekítve 204 terabyte adatot jelent, mely tárolása jelenleg szinte megoldhatatlan. Éppen ezért szükséges egy olyan tárolási módszert kidolgozni, melyben csak a szükséges értékek kerülnek letárolásra.

Az egyik legnagyobb további probléma, hogy a matematikai maximalizálás nem feltétlenül célravezető a felhasználói elégedettség maximalizálásakor: lehet, hogy az

adott felhasználó éppen a legnépszerűbb tartalmakat kedveli, vagy csak az általa követett, barát-előadók zeneszámain/tartalmait látogatja szívesen. Ennek megoldására az ajánlási lista utószűrése adja a megoldást.

5.5 Előszűrési feladatok

5.5.1 Jogilag vagy etikailag mellőzendő tartalmak kiszűrése

A rendszer egy világszinten alkalmazható megoldás prototipikus modellje. Ennek megfelelően bizonyos - a dolgozatban nem részletezett – törvényi és etikai előírások vonatkoznak rá, ráadásul ezek területenként, népcsoportonként változóak lehetnek. Ennek megfelelően szükség van olyan eljárásra, mely ezeket a megszorításokat kezeli, elkerülve ezáltal a későbbi kellemetlenségeket.

5.5.2 Felhasználói vélemények súlyozása közösségi ajánlásnál

A közösségi ajánlás a rendszer egyik legfontosabb funkciója. Ennél kiemelten fontos, hogy a valós szórakozóhelyek (vagy online tartalomszolgáltató rendszerek) bízzanak a szoftverben, merjék használni. E bizalom eléréséhez pedig mindenképpen szükséges az, hogy az adott szolgáltató egység beleszólhasson (legalább bizonyos fokú súllyal) a tartalomlistába, nehogy olyan elem kerüljön bele, melyet nem kíván szolgáltatni.

Megállapítható továbbá, hogy a szolgáltatóipari egységek szeretnek kifejezett figyelmet biztosítani régebbi, rendszeren visszajáró tagjaik, törzsvendégeik számára. Ennek megfelelően biztosítandó az, hogy a minél többet tagként ott tartózkodók nagyobb súllyal számíthassanak a lista generálásakor, mint az újonnan érkezők. Emellett biztosítani kell a szórakozóhely vagy egyéb egység számára az alapvető profil lehetőségét: a szolgáltatóipari helyek szeretik, ha leírhatóak, jellemezhetőek a nagyközönség számára, a profil ezt reprezentálja.

5.5.3 Mellőzött tartalmak kiszűrése

A közösségi szolgáltatáson keresztül a felhasználónak lehetősége van megadni olyan tartalmakat, melyeket kifejezetten mellőz, és semmiképpen sem szeretne az ajánlási listában viszont látni. Algoritmikai szempontból ezek az elemek legjobb, ha bele sem kerülnek a lehetőségek közé, legfeljebb negatív hatásuk érzékelődik a hozzájuk hasonlókon.

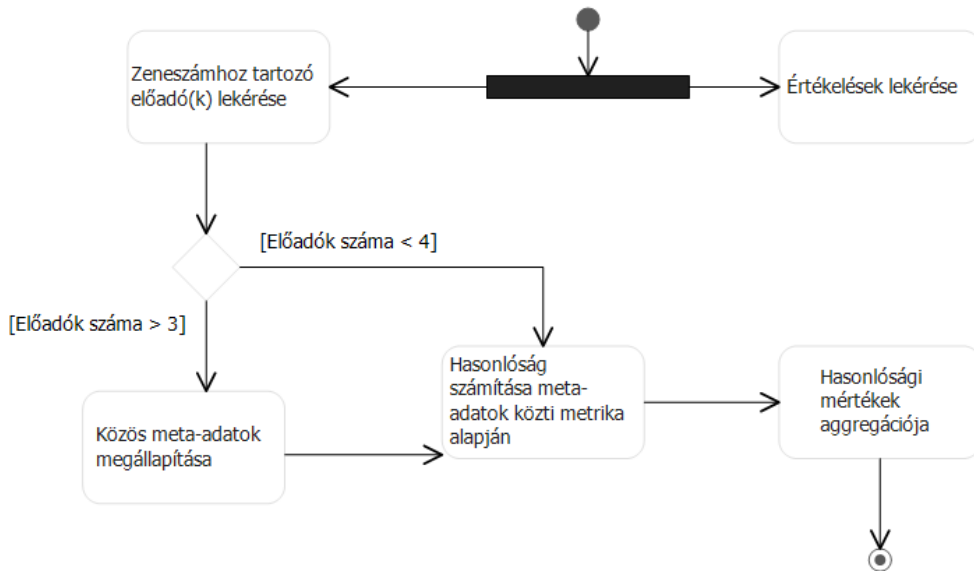
5.6 Algoritmikai feladatok, alkalmazott megoldások

Az algoritmikus fázis végzi az ajánlórendszer legfontosabb feladatait: megállapítja azon kapcsolatokat, illetve a kapcsolatok értékeit, melyek az ajánlási listát képzik. Ebben a fázisban kialakul egy kezdetleges ajánlási lista, mely az utószűrési fázisban nyeri el végleges formáját.

5.6.1 Hasonló tartalmak megadása

A tartalmak hasonlóságának megadása a metrikáknál leírt módon valósul meg: a felhasználói értékelések és a meta-adatok hasonlósága együttesen számít. A meta-adatok közti metrikát felhasználó algoritmus lépései a következők:

1. **Zeneszámhoz tartozó előadók megállapítása:** lekérdezésre kerülnek a zeneszám előadói. Fontos megjegyezni, hogy egy zeneszámhoz több előadó is tartozhat.
2. **Előadók számának lekérése:** amennyiben az előadók száma legalább 4, úgy kizárólag az előadók közös meta-adatai számítanak. Erre azért van szükség, mert a nagyon sok meta-adattal rendelkező zeneszámok előnyben részesülnének.
3. **Hasonlóság számítása:** a meta-adatok közti metrika aggregációja alapján kialakulnak a zeneszámok közti kapcsolatok értékei.
4. **Hasonló zeneszámok redukálása:** a rendszerben csupán 10 darab hasonló zeneszám letárolására van szükség, ennek megfelelően a felesleges kapcsolatok törlésre kerülnek.

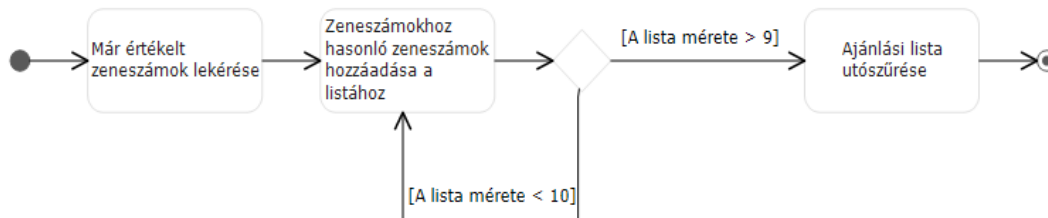


14. ábra: Zeneszámok közti hasonlóság megállapítása

5.6.2 Ajánlás individuális felhasználók számára

Az individuális felhasználók számára az ajánlás a tartalom alapú hasonlóságokra épül, melyek az előző (5.6.1) pontban kifejtetteknek megfelelően felhasználja mind az értékeléseket, mind a meta-adat hasonlóságokat. Az algoritmus lépései a következők:

1. **A felhasználó által már értékelt zeneszámok lekérése:** lekérésre kerül minden, a felhasználó által eddig értékelt zeneszám.
2. **A már értékelt zeneszámokhoz hasonló zeneszámok lekérése:** lekérésre kerül minden, a már értékelt zeneszámokhoz hasonló zeneszám. Ez addig folytatódik, ameddig nem lesz legalább 10 eleme a listának.
3. **Ajánlási lista utószűrése:** törlésre kerülhetnek az eddig értékelt zeneszámok, a felesleges kapcsolatok, és végrehajthatóak az egyéb utószűrési feladatok.



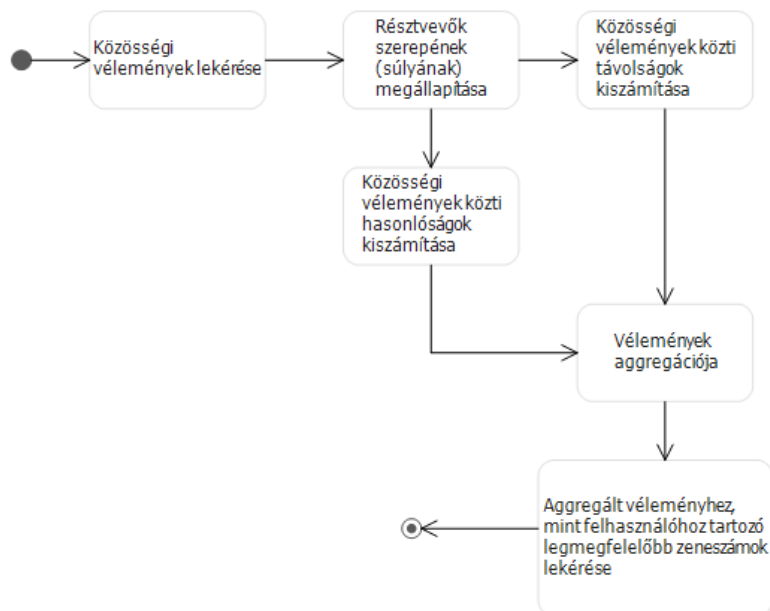
15. ábra: Ajánlás individuális felhasználók számára

Az eddig meghallgatott zeneszámok az utószűrésben kikerülhetnek a listából, de akár a lista részeit is képezhetik.

5.6.3 Ajánlás közösségek számára

A közösségek számára történő ajánlás a (4.) fejezetben bevezetett módszeren alapszik.

1. **Közösségi vélemények lekérése:** lekérdezésre kerülnek a közösséget aktuálisan alkotó közönség véleményei.
2. **Résztevők súlyainak megállapítása:** megállapításra kerül a szórakozóhely szerepe (súlya), valamint az esetleges törzsvendégek.
3. **Közösségi vélemények közti távolságok és hasonlóságok számítása:** kiszámításra kerülnek a megfelelő értékek a (4.) fejezetben definiáltak szerint.
4. **Vélemények aggregációja:** a (4.) fejezetben meghatározott, a dolgozat által szintetizált képlet felhasználásával megállapításra kerül a közösségi vélemény, mellyel már individuális felhasználóként megadható az ajánlási lista.
5. **Ajánlott lista lekérése:** az ajánlott lista így már, mint individuális felhasználó számára kérhető le.



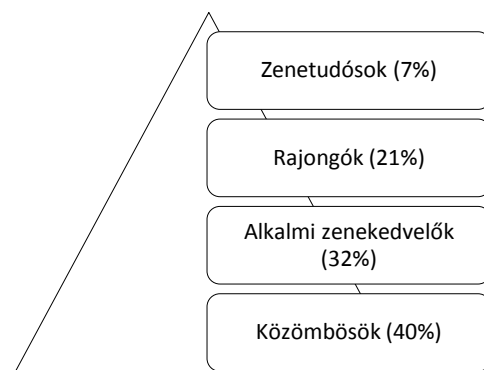
16. ábra: Közösségi ajánlás folyamata

5.7 Utószűrési feladatok

Sok ajánlórendszerrel rendelkező tartalomszolgáltató rendszernél figyelhető meg az a probléma, hogy a felhasználó értékeléseiből kikövetkező tartalmak egyszerűen nem érdeklik a felhasználót. Ez nem azért van, mert az ajánlási algoritmus hibásan maximalizálja a hasznosságfüggvényt, hanem azért, mert a modern tartalomszolgáltató rendszerek gyakran nem csak egy típusú tartalmat kínálnak.

5.7.1 Elemek sorrendje és aránya az ajánlási listában

Megfigyelhető például az, hogy a közösségi hálózatok népszerűségének növekedésével egyre több tartalomszolgáltató rendszer támaszkodik ezen hálózatok közönségeire, és ezen hálózatok által tárolt adatokra. Ennek megfelelően megvalósításra kerültek olyan funkciók, melyek kihasználják a felhasználók kapcsolatait, sokszor van lehetőség például megtekinteni a barátok által előállított tartalmat. Emellett a lehetőség mellett fontos még megemlíteni azt is, hogy bizonyos felhasználók kevesebbet foglalkoznak az adott tartalommal tudományos szinten (például rendszeresen előállítják azokat), míg bizonyos felhasználók csak fogyasztják azokat. Erre jó példa egy együttes zenésze és a koncertjükön résztvevő zenekedvelő különbsége. A felhasznált irodalom által leírt felhasználói típusok (zenetudós, rajongó,



alkalmi zenekedvelő közömbös) a rendszer által legfontosabb, 16-45 éves korosztálybeli eloszlása a következő [7]:

Ezek figyelembe vételével megállapítható az, hogy a tartalomszolgáltató rendszerben található adatok klasszifikálásra szorulnak azok típusai szerint. A tervezett tartalomszolgáltató rendszerben ezek az alábbiak: *követett/barát előadók zeneszámai, népszerű tartalmak* (a hosszú farok elejéről), *ismeretlen ajánlott tartalmak* (általában a hosszú farok hátsó részéről), valamint *már előzőleg is értékelt, ismert tartalmak*. Bizonyos felhasználók egyiket jobban szeretik, mint a másikat, így az értékelések félrevezethetőségét elkerülendő bevezetésre kerül az ajánlási lista klasszifikálása és utószűrése.

Amennyiben egy felhasználó nagyrészt az egyik típusú tartalmat fogyasztja szívesen, úgy az idő előre haladtával ajánlási lista egyre nagyobb részét teszi ki, de semmiképpen sem uralja a teljes listát: meg kell adni a lehetőséget a felhasználónak a véleményváltásra

5.7.2 Normalizálás: a hanganyagok egyenlőségének megtartása

A zeneszámok hangerejének normalizálása több szempontból is fontos feladat. Egyrészt a zeneszámok különböző hangerőkön kerültek felvételre, másrészt az újabb zeneszámok nagymértékben komprimálásra kerülnek, mely a dinamikatartomány tulajdonképpeni összenyomása. A túlzott kompresszorhasználat következtében a zeneszám hangereje sokkal erősebbnek érződik, mely a modern zeneszámok kedvelt tulajdonsága, azonban ezáltal egymás mellé téve egy régebbi és egy újabb zeneszámot a kettő nem illik össze.

Ha normalizálás nélkül kerülnek lejátszásra a hanganyagok, az még egy fontos problémát eredményez: a felhasználóknak általánosan jobban tetszik az a zeneszám, ami hangosabb, így ez torzítaná az értékeléseket, amire az algoritmus nagymértékben támaszkodik.

A normalizálás a következőképpen zajlik [10]:

Legyen az adott zenei jel $x[n]$, mely a erősítéssel rendelkezik:

$$\hat{x}[n] = ax[n]$$

Ekkor $|\hat{X}[k]|$ spektrum is skálázódik a szerint:

$$|\hat{X}[k]| = a|X[k]|$$

Ekkor a normalizálás egy konstans segítségével történik:

$$20\log_{10}(|\hat{X}[k]|) = 20\log_{10} a|X[k]| = 20\log_{10}(a) + 20\log_{10}|X[k]|$$

A normalizálás tulajdonképpen a jel kiskálázása egy adott decibeltartományra. Az adott algoritmus először megkeresi a legnagyobb kitérés értékét (peak), majd megállapítja, hogy ez milyen távol van a kívánt értéktől. Ezután ezzel a távolságértékkel csökkenti vagy növeli a teljes hanganyag decibelértékét.

5.8 Algoritmikus módszerek: ajánlás a tervezett rendszerben

A tervezett rendszer egy gráf alapú, kapcsolatorientált rendszer. Ez számos problémára nyújthat megoldást, azonban az ajánlóalgoritmus legnagyobb erőssége abban rejlik, hogy - egyrészt a már előzőekben is említett ajánlási lista sorrendjének és tartalomtípusának személyre szabásával nem feltétlenül a matematikai függvényt, hanem a felhasználó elégedettségét próbálja maximalizálni, másrészt a klasszikus – például mátrixfaktorizálást – használó megoldásoknál radikálisan, nagyságrendekkel gyorsabb úgy, hogy a pontosságból nem áldoz fel sokat. Természetesen azzal, hogy a metrikák ismertek, az ajánlási feladat még nem oldódott meg: nem adható mindig ugyanaz a néhány szomszéd tartalom. Az ajánlási feladatok és az azt megoldó módszerek a következők:

5.8.1 Felhasználói profilalkotás: a gráf inicializálása I.

A hidegindítási probléma és a minél pontosabb és változatosabb felhasználói profilok megalkotása érdekében a rendszer a regisztráláskor végigvezeti a felhasználót két nagyon lényeges lépésen (például felhasználónév, jelszó):

1. **Profiltípus kiválasztása:** a rendszer célja kiszolgálni mind individuális felhasználók (zenekedvelők) igényeit, mind szórakozóhelyek közönségeit, és emellett célja az amatőr előadók műveit is terjeszteni. Ezen célok eléréséhez különböző profilok szükségesek, így a rendszer regisztrációkor felkínál három lehetőséget: „egyszerű” felhasználó, előadó – a zenei karrier fellendítéséhez, valamint szórakozóhely. A különböző profilokhoz különböző lehetőségek társulnak, például egy előadónak lehetősége van zeneszámai feltöltésére és

tagelésére, egy szórakozóhelynek pedig lehetősége van felhasználókat fogadni, és közösségi ajánlást kérni számunkra. Ezen kívül a rendszer közösségi hálójában is szerepe van a profiloknak: a szórakozóhely például megoszthat követői számára releváns reklámokat, később pedig akár bevezetésre kerülhetnek események is, mint például péntek esti rendezvény a szórakozóhelyen, vagy egy előadó koncertje.

2. **Profil kalibrálása:** ezen a ponton a felhasználónak lehetősége van megadni az általa kedvelt (és mellőzött) stílusokat, előadókat, zeneszámokat egy erre a célra kifejlesztett mátrix-elrendezésű felületen. Az előadónak lehetősége van megadni hozzá kapcsolódó meta-adatokat, illetve más, hozzá hasonló előadókat. A szórakozóhelynek lehetősége van megadni egy általános zenei profilt, mely az alapvető célközönséget is szemlélteti. Ezen a helyen lévők bizonyos arányban tudnak változtatni, azonban a rendszer szeretné elkerülni azt a problémát, hogy kevés – a rendszert használó – felhasználó képes legyen megváltoztatni egy szórakozóhely teljes zenei profilját.

Jól látható, hogy a lépések nem vesznek igénybe feltétlenül sok időt, és nagyon hasznosak egy alapvető profil megadásához. Ez alapján már mindhárom típusú felhasználó elkezdheti élvezni a rendszer által nyújtott szolgáltatásokat. Természetesen egy szórakozóhely más meta-adatokkal rendelkezik, mint egy felhasználó.

5.8.2 Zenei profilalkotás: a gráf inicializálása II.

Az ajánlórendszer feladata, hogy a felhasználói profilokhoz megfelelő zenei profilokat társítson. Ehhez a felhasználói profilok mellett szükséges a megfelelő zenei profil, melyet a zeneszámot leíró meta-adatok jellemeznek. A zeneszám meta-adatainak megadására azok feltöltésekor mindenképpen sor kerül (kötelező megadni legalább 3-at), azonban később ez a lista még bővíthető. A rendszer a lista kitöltésekor felajánlja a beírt meta-adatokhoz aktuálisan legjobban kapcsolódókat (leghasonlóbbakat), ezzel is növelve a tagelés mértékét. A zeneszámokat jellemző

meta-adatok ezután bekerülnek a hasonlósági gráfba, a köztük lévő kapcsolat pedig az értékelésekből következik (részletesebben a következő pontban).

6 A megvalósított modell és eredményei

A fejezet bemutatja a rendszer alapvető felépítését, a hallgatók szerepköreit a rendszer megvalósításában, az alkalmazott tesztkörnyezetet, valamint a kapott megoldásokat. Bemutatásra kerülnek továbbá a piacon jelenleg elérhető legjobbnak mondott hozzáférhető rendszer, az Apache Mahout eredményei, melyek összehasonlításra kerülnek a megvalósított, kezdeti saját modellel.

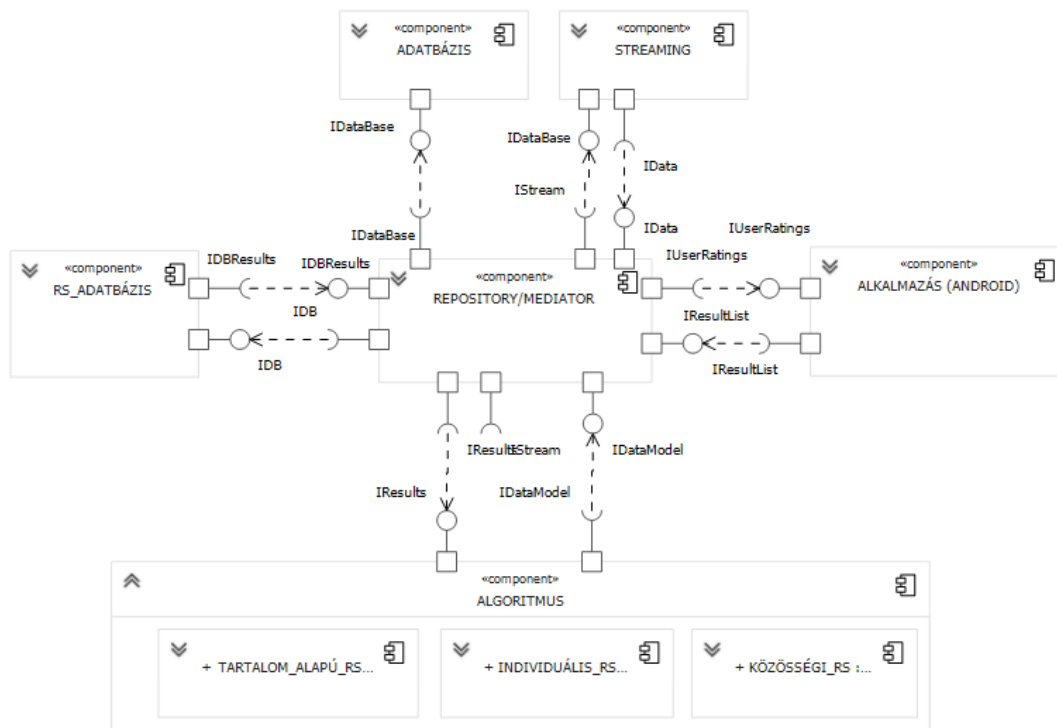
6.1 Rendszeranalízis

A rendszertervezés/rendszeranalízis a szoftvertermék létrehozására irányuló tevékenységek és azok eredményeinek **teljes** folyamata. Egy megfelelő terv tehát tartalmazza az ötlet kidolgozásától a kész rendszer karbantartásáig az összes részletet, azonban a dolgozat ebből a folyamatból kizárólag azokat a részeket emeli ki, melyek a rendszer átlátásához és tesztelésének megértéséhez szükségesek.

A rendszer egyébként a szoftvertervezésnek megfelelően, evolúciós prototípusként készült két másik hallgató, Klicsu Zsolt és Sallai Gergely bevonásával, akikkel a napi szintű kommunikációnak és megfelelő projektmenedzsmentnek köszönhetően létrehozásra kerülhetett e nagy rendszer kezdeti modellje.

6.1.1 A rendszer alapvető felépítése

Mint minden rendszernél, így a létrehozott prototipikus modellnél is nagyon fontos az alapvető felépítés, komponensekre bontás. A rendszer alapvetően 4 fő komponensből áll: a tárolásért felelős adatbázisból, az algoritmikus számításokat végző algoritmus komponensből, az adatátvitelt végző streaming részből, valamint a kliensből. A dolgozat ezek közül az algoritmikus számítást végző komponensre koncentrálna.



18. ábra: A prototipikus modell komponensdiagramja

Ahogy az az ábrán is látható, a tervezett rendszer a repository architektúráis tervezési mintát alkalmazza, melynek lényege egy központi rendszerelem (a tároló vagy tárhely), mely összeköti a rendszerkomponenseket. Rendkívül nagy előnye, hogy az egyes alrendszerek külön-külön is alkalmazhatóak, nem „ismerik” egymást, csupán az interfészeket, melyeken keresztül kommunikálnak az összefogó rendszerelemmel.

6.2 A tesztkörnyezet

A megvalósított rendszer az alábbi számítógépen került tesztelésre:

- **Processzor:** Intel Core i7-2600 CPU @ 3.40 GHz
- **Telepített memória (RAM):** 8,00 GB
- **Operációs rendszer típusa:** Microsoft Windows 7 64-bites operációs rendszer

A megvalósított rendszer **tárolási megoldásként** Microsoft SQL Server-t használ, a feladatok jó részének megoldása is ebben valósult meg. A megvalósított rendszer a piacon jelenleg első, ingyenes megoldással, az **Apache Mahout** legfrissebb verziójával került összehasonlításra, melynek szüksége volt MySQL Server 5.2-re, valamint Java JDK 7u9-re (mindenből a legfrissebb stabil verzió került telepítésre).

6.3 Tesztadatok, tesztesetek és eredmények

A tesztesetek elsődlegesen úgy kerültek kialakításra, hogy minél jobban illeszkedjenek a dolgozat célkitűzéseire és feladataihoz, de fontos szempont volt, hogy a rendszer összemérhető legyen más, a piacon jelenleg megtalálható, hozzáférhető megoldással is. Ennek megfelelően a tesztesetek lefedik a személyre szabott ajánlás kalkulálását individuálisan egy, illetve az összes felhasználóra, valamint a zenei hasonlóságok megállapítását egy-egy zeneszám párosra, és a jelenleg digitálisan forgalomban lévő összes zeneszámmra is.

Az **alkalmazott adathalmaz** a „Last.fm Million Song Dataset”, melyből a felhasználók hallgatási szokásai kerültek kinyerésre, valamint a MusicBrainz adatbázisában lévő adatok, melyek a jelenleg összes digitálisan kiadott zeneszám adatait tartalmazzák. Az értékelési/hallgatási adatokból 10 millió sor került beszúrásra, a többi – MusicBrainz-ről – elérhető adat nagy részben letárolásra került (a tárolási modell Klicsu Zsolt kapcsolódó dolgozatában tekinthető meg).

6.3.1 A megvalósításból adódó, rendszerek közti eredendő különbségek

Nagyon fontos megemlíteni, hogy bár a két rendszer célja közel azonos, az összehasonlítás nem lehet teljesen mérvadó, hiszen nemcsak az alkalmazott megoldási módszer más, hanem a megvalósítás teljes folyamata. Az Apache szoftverénél például elkerülhetetlen volt az adatok fájlba mentése, vagy memóriába töltése, mert az adatbázisból olvasás olyan nagy mértékben lassította a folyamatot, hogy a feldolgozási idő több napra nyúlt ki. Ennek megfelelően a tervezett rendszer adatbázisból nyeri az adatokat, míg az Apache megoldása közvetlenül a memóriából. További különbség, hogy az Apache megoldása RMSE-hibákból tanul, míg a tervezett rendszer alapvető tanítása a felhasználó visszajelzéseken, illetve különböző közösségi kapcsolatinformációkon alapszik. Így az algoritmusok pontossága már-már matematikailag összemérhetetlen, így javasolt egy esetleges A/B teszt, felhasználók, esetleg szórakozóhelyek bevonásával.

6.3.2 Megoldandó feladatok, választott megoldások

Az ajánlórendszer elsődleges feladata egy olyan prototipikus modell elkészítése volt, mely képes individuális felhasználóknak zeneszámokat ajánlani, valamint az értékelésekből tanítható, esetlegesen pedig képes kiszolgálni szórakozóhelyek valószínűségeit is. Az Apache legjobb megoldása saját bevallása szerint erre az ALS-WR mátrixfaktorizáló algoritmus, azonban a dolgozat keretében tesztelésre került a tartalom-alapú, Pearson-korrelációs metrikát alkalmazó megoldás is. A tervezett algoritmus a (6.) fejezetben bevezetett metrikákkal és megoldásokkal került tesztelésre.

Nagyon fontos kiemelni, hogy az Apache Mahout tartalmaz megvalósításokat adatbázisban tárolt adatok használatára, azonban mivel a feladatokat egyesével hajtja végre, a sok olvasás annyira lelassítja a rendszert, hogy **az Apache rendszerének teszteléséhez az adatok a memóriába kerültek betöltésre**. Ez természetesen nagyon nagy adathalmazzal megvalósíthatatlan lenne, de a 10 milliós adathalmaznál ez még nem okozott problémát.

6.3.3 Az Apache Mahout eredményei az alkalmazott tesztalmazon

1. Eset: adott zenei tartalomhoz 10 leghasonlóbb zenei tartalom megadása

Tartalom azonosítója	Számítási idő
228	219 ms
229	604 ms
230	194 ms
231	140 ms
232	133 ms
233	404 ms
234	295 ms

1. táblázat: Részlet az Apache Mahout hasonlóságszámításaiból

Megfigyelhető, hogy a számítási idő azonosítónként jelentős különbséget ad. Ez abból adódik, hogy míg bizonyos elemek sok értékeléssel rendelkeznek, addig más elemek csak egy-egy értékelést kaptak összesen. Az időkülönbségben az is közrejátszik, hogy az adott elemek értékelései mennyire származnak ugyanattól a felhasználótól. Kimérhető, hogy az átlagos számítási idő körülbelül **194 milliszekundum**, melyet felszorozva a tesztadatok tartalomszámával (**12213568 darab**) megkapható, hogy a teljes számítás nagyjából **27 napot** venne igénybe. Ez az adat elsőre rendkívül soknak tűnik, azonban megjegyzendő, hogy ezen számítások elvégzése csupán akkor szükséges, amikor a rendszer kezdeti indítása zajlik, valamint akkor, ha új elem kerül a rendszerbe. A becslés azért mérhető a szorzással, mert a Mahout minden egyes számítást külön, egymás utáni sorrendben végez el.

Nagyon fontos megjegyezni, hogy a tesztelt 10 millió sornyi adatban nincs értékelés mind a 12213568 zeneszámra (eleve több a számuk, mint az adatok sorszáma), így az ajánlás 27 napnál jóval kevesebb idő alatt lefutna, de a zeneszámoknak csak töredékét lenne képes ajánlani.

Megállapítható tehát, hogy az Apache megoldása kész rendszereknek hasznos lehet, azonban a tervezett rendszerbe tölthetők fel zeneszámok, melyekből akár percenként is érkezhetnek újak. Megjegyzendő továbbá, hogy az Apache tesztelt megoldása **kizárólag a felhasználói értékeléseken alapul**, így nem oldja meg a hidegindítási

problémát, ráadásul az alkalmazott tesztalmazban nem is található minden zeneszámhoz értékelés (csak 10 millió sornyi adat került beillesztésre), így bizonyos zeneszámokhoz **lehetetlen a tesztelt adatok alapján ajánlani**.

2. Eset: adott felhasználónak történő 10 legjobb elem ajánlása

Felhasználó azonosítója	Számítási idő
97	3999 ms
98	10899 ms
99	9582 ms
100	1931 ms
101	4133 ms
102	5435 ms
103	30043 ms

2. táblázat: Részlet az Apache Mahout mátrixfaktorzással történő predikciójából

Az előző méréshez hasonlóan megfigyelhető a számítási idők elég nagy szórása. Azért fordulhatnak elő ilyen nagymértékű különbségek, mert egy-egy felhasználó kevés, míg más felhasználók kifejezetten sok tartalmat értékeltek. Kimérhető, hogy az átlagos számítási idő körülbelül **13085 milliszekundum**, melyet felszorozva az összes felhasználó számával (**69878 darab**) megkapható, hogy a teljes számítás **nagyjából 10 napot** venne igénybe. Ez azt jelenti, hogy az újonnan bekerülő felhasználók csak 10 naponta kaphatnának ajánlott tartalmat, ráadásul a jelenlegi felhasználók értékelései is csak 10 nap után tanítanak a rendszert, mely egy valós időben működő rendszer esetén nem kielégítő.

Megjegyzendő továbbá, hogy a megoldás kizárólag a 10 millió sornyi adattal került tesztelésre, és egy nagy tartalomszolgáltató rendszerben ennyi adat egy hét alatt is könnyedén keletkezhet.

Az előző eseteken kívül általánosan elmondható, hogy az Apache megoldása működéséből adódóan is alkalmazhatatlan nagy felhasználószámú tartalomszolgáltató rendszer esetén, hiszen a napi értékelésekre és egyéb adatokra szükség van nemcsak az adott alkalmi számításhoz, de a későbbiekhez is. Egy ilyen tábla következményeit

nem e dolgozat hatásköre megítélni, azonban az elmondható, hogy a rendszer az idő elteltével egyre lassabb lesz, mely egy szolgáltató esetén akár végzetes is lehet.

6.3.4 Saját megoldás eredményei az alkalmazott teszthalmazon

A táblázatok rövidsége és mássága miatt fontos megemlíteni a tesztek elején, hogy míg az Apache szoftvere az előfeldolgozási és számítási fázis után egyesével számolja megoldásait, addig a tervezett rendszer a gráf-alapú, SQL-ben megírt megvalósításnak köszönhetően **egyszerre** képes minden megoldást megadni. Fontos megemlíteni, hogy míg az Apache a memóriában tárolt adatokból számolt, addig a saját megoldás a merevlemezről, a rendszerhez külön dolgozatban kifejlesztett

1. **Eset: Adott zenei tartalomhoz 10 leghasonlóbb zenei tartalom megtalálása** a közösségi szavazások alapján minden zeneszámmra: 6 másodperc
2. **Két zeneszám közti hasonlóság kiszámítása tartalom alapú ajánlással**, a meta-adatok felhasználásával: 600 milliszekundum
3. **Eset: zenei tartalomhoz 10 leghasonlóbb zenei tartalom megtalálása a tartalom alapú hasonlóság alapján:** 14 másodperc
4. **Eset: felhasználónak történő 10 legjobb elem ajánlása:** 777 milliszekundum

adatbázisból nyerte az adatokat [21]. A **reprodukálható** mérési eredmények a következők:

Nagyon fontos megállapítás, hogy a megvalósított megoldás a gráf alapú reprezentációnak és az algoritmusok megvalósítási módjának köszönhetően az ajánlás **egyszerre** is történhet, így nem szükséges elemenként iterálva elvégezni az adott műveleteket. Az alkalmazott, korábbi fejezetben bevezetett módszereknek köszönhetően **az algoritmus futási ideje nagyságrendekkel alacsonyabb**, mint a jelenlegi megoldások, igaz, hogy működésben és megvalósításban is jelentős különbségek vannak. Bár az összehasonlítás nem teljesen releváns, a kifejlesztett

algoritmus a piac egyik nagy, meghatározó szereplőjével került összehasonításra, és eredményei impozánsak.

6.3.5 Saját megoldás és az Apache Mahout eredményeinek összehasonlítása

Nagyon fontos ismét kihangsúlyozni, hogy a dolgozat megoldása az adott problémákra teljesen más aspektusból közelíti meg a problémát, mint a vizsgált Apache Mahout, azonban a piaci termékként való funkcionalitás, mint cél miatt mindenképpen szükséges volt a jelenlegi rendszerrel összemérni.

Megfigyelhető volt az Apache Mahout mérésekor, hogy az elemeket – még ha párhuzamosított megoldással is, de **egyesével dolgozza fel**. Van tehát egy előfeldolgozási fázis, mely az adatbetöltés és faktorizálás, majd egy ajánlási fázis, melyben az egyes ajánlások kerülnek kiszámításra. A saját rendszerben azonban ezzel ellentétben az adott feladatra minden eredmény egy-egy eljárásban valósul meg, **egyszerre történik a számítás/szűrés**, így a lekérdezés ideje nem változik lényegesen 1 felhasználónak, vagy 1.000.000 felhasználónak történő ajánláskor sem (természetesen a fetch időt nem számítva).

Továbbá megfigyelhető, hogy az egyedi, gráf alapú megoldás és saját metrikák alkalmazásával **az eredmények nagyságrendekkel gyorsabban megadhatók**, így a tervezett rendszer **alkalmas lehet a mai népszerű, modern szolgáltató rendszerekben történő valós idejű alkalmazásra** is, mely a dolgozat alapvető célkitűzése volt.

Nagyon fontos még megemlíteni, hogy az ajánlás hibájának mérése annyira különböző lenne, hogy értelmetlen kimérni, azonban mindkét rendszer tanítható a kellő pontosság eléréséig, a tanításnak csupán ideje és módja különböző.

6.3.6 További tesztelés, összehasonlítás komplett piaci megoldásokkal

A dolgozat keretein belül sajnos nem lehetséges összehasonlítani a teljes piaci megoldást más rendszerekkel (mint például a Pandora), de a dolgozat javaslatot tesz a rendszer későbbi piaci tesztelésére: előre láthatólag a jelenlegi prototipikus modell egy kiegészített verziója tesztelésre kerül néhány szórakozóhely és néhány száz

felhasználó bevonásával, melynek eredményei alapján elkészülhet a rendszer véglegesnek tekinthető változata. Ezután a rendszer bevezetésre kerülhet nagy, valós szórakozóhelyeken, kiszolgálva ezek közönségeit akár sokmillió felhasználó esetén is. A tervezett teszt egy A/B teszt lesz, melynek során a jelenlegi piaci megoldások által adott ajánlási lista kerül összehasonlításra a dolgozat keretein belül elkezdett prototipikus modellel.

7 Konklúzió, további feladatok

Az ajánlórendszerek területén tett kutatás multidiszciplináris. A mesterséges intelligencián, adatbányászatban belül is sok területet érint, de elmondható, hogy a mai infokommunikációs területek határain helyezkedik el. Vizsgált területei közé tartoznak a szűrési módszerek, a közösségi kapcsolatok, az adott tartalom személyre szabása, a tanítható hálózatok, a jelfeldolgozás, és sok más terület is, továbbá egyértelmű, hogy e területeknek hatalmas piaci hatása van, így nem lehet eltekinteni a gazdasági és társadalmi vonatkozásokról sem.

Ez a dolgozat elsősorban egy rendszer prototipikus megvalósítására irányult, összefoglalta az ennek megvalósításához szükséges ismereteket, elemezte majd alkalmazta azokat kifejezetten média- illetve zenei tartalmakra. Elmondható továbbá, hogy a dolgozat elsődleges témája (az ajánlási predikciós probléma) kiegészült egy modern területtel: a közösségi kapcsolatok felhasználásával történő csoportos és egyéni ajánlással.

A dolgozat célkitűzése volt a kutatás mellett egy termékként is helytálló prototipikus modell megtervezése és elkészítése, melynek alapvető evolúciós modellje elkészült, eredménye más, régóta fejlesztés alatt álló, piaci súllyal rendelkező rendszerrel összemérve is kiemelkedő. Ennek eléréséhez a dolgozat nemcsak az információs technológia terén, de a társadalmi és gazdasági területeken is vizsgáldott, felhasználva ezek mai meglátásait, mint például a hosszú farok modellt.

7.1 A célkitűzések verifikálása: a dolgozat eredményei

A dolgozat a bevezetésben előterjesztett néhány szempontot, melyekből kialakult a prototipikus modell alapvető célja, mely célok eléréséhez számos feladatot – és részfeladatot kellett megoldani. Ezek megoldásaként a dolgozat elsődleges eredményei a következők:

- **Jól skálázódó, akár sokmillió felhasználó- és tartalomszámra alkalmazható ajánlóalgoritmus**, mely figyelembe veszi az adott tartalom fajtájának egyedi feladatait, nehézségeit.

- **Valós idejű, adott individuális felhasználó számára történő ajánlás**, mely ajánlás a felhasználó visszajelzései alapján úgy módosul, hogy minél jobban kielégítse a felhasználói igényt, nagyobb elégedettséget okozva ezáltal.
- **Valós idejű, valós közösségekből szerveződő csoportok számára történő ajánlás**, mely figyelembe veszi az adott közösségben tartózkodók társadalmi szempontjait, valamint a szórakozóhelyek, mint speciális közösségszervező helyek igényeit úgy, hogy a módszer piaciilag is helytálló legyen.
- **Valós piaci igényeket kielégítő, termékként is helytálló megoldás**, mely nemcsak zenei, de akár más tartalomszolgáltató rendszerek által is használható, sőt, sokszor használhatóbb, mint a piac jelenlegi megoldásai.
- **Modern szempontokat figyelembe vevő rendszer**, mely nemcsak az egyszerű hasznosságfüggvényt maximalizálja, de olyan egyedi szempontokat is figyelembe vesz, mint a különböző típusú tartalmak, vagy felhasználói szokások.

7.2 A felmerülő további feladatok megoldásai

A dolgozatban leírt szolgáltató rendszer fejlesztése közben számos megoldandó alfeladat merült fel, melyek kidolgozása akár egyenként is érdekes téma lehetne egy hasonló szintű dolgozat számára. Bár a dolgozat kereteit ezek részletes kidolgozása meghaladja, a következő összefoglaló megpróbálja leírni a fejlesztés közben felmerült kérdéseket, és azokra adott válaszokat:

- **Predikciós modell kérdése:** nagyon fontos lépés volt a fejlesztésben, hogy a dolgozat elszakadt a jelenleg alkalmazott, méltán népszerű, nagymértékben kidolgozott klasszikus ajánlási módszerektől, és egy olyan, gráf-alapú megoldást javasolt, mely mint kiderült bizonyos szempontból alkalmasabb lehet a felhasználók kiszolgálására.
- **Hibrid módszer kidolgozása:** megfigyelhető, hogy a kollaboratív szűrés, valamint a tartalom alapú ajánlás egyaránt megtalálható a rendszerben. Ennek következtében az alkalmazott módszer számos olyan klasszikus problémára ad

megoldást, mint például a hidegindítási probléma. Ezek a problémák évekig lassították az ajánlórendszerek működését és korlátozták azok hatékonyságát.

- **Zeneszámok, mint tartalmak problémái:** nagyon fontos kiemelni, hogy számos olyan probléma merült fel a fejlesztés közben, ami kifejezetten zeneszámokra jellemző. Ilyen probléma volt például a tartalmak hasonlóságának megállapításánál az, hogy sok zeneszámnak van több előadója, melyek így előnyt élveznek a jellemző attribútumok terén. Ezen problémák megoldása meglátszik a teszteléskor létrehozott ajánlási listákon, sokkal felhasználóbarátabbá téve azokat.
- **Dinamikus adatok feldolgozása:** a tesztek elvégzésekor jól látható volt, hogy rendkívül nagy feladat megbirkózni azzal, hogy a rendszerben szereplő adatok dinamikusak: törlésre kerülhetnek meglévő adatok, illetve új adatok kerülhetnek be, melyek megkövetelik a gyors és megfelelően reagáló ajánlást.
- **Közösségi ajánlás speciálisan szolgáltatóipari egységekre:** bár a közösségi vélemények leírásának és azok hasonlóságának, illetve távolságának megoldásai már részben kidolgozásra kerültek, a dolgozat által javasolt apró változtatással még pontosabbá tehető a közösségi ajánlás. A dolgozat megvizsgálta a csoportos ajánlást speciálisan szolgáltatóipari egységekre, melyeknél a csoportos döntéshozásban szereplő adatok dinamikusan kerülnek feldolgozásra, így nagyban eltér a klasszikus problémáktól.

7.2.1 A rendszer piaci termékként való alkalmazhatósága

Külön kiemelésre kerül, hogy a tervezett prototipikus modell úgy került megvalósításra, hogy alkalmazható legyen akár piaci termékként, ahol az általános feltételekhez hozzá társulnak egyéb funkcionális és nem funkcionális igények is. Megfelelő felhasználói teszteléssel, és a felhasználók, illetve szolgáltatóipari egységek véleményeinek feldolgozásával, a rendszer további fejlesztésével a modell alkalmas teljesen kész, piaci termékként megállni a helyét a modern informatikai gazdaságban.

- **Előszűrési lehetőség:** a dolgozatban – és a fejlesztésben – külön kiemelésre került egy olyan előszűrési lehetőség, mely a modell piaci alkalmazhatóságát segíti. Ebben az előszűrési fázisban megadhatóak olyan jogi vagy etikai megszorítások, melyek egy terméknel alapkövetelménynek számíthatnak.
- **Utószűrési lehetőség:** bár a dolgozat nem tér ki részletes gazdasági elemzésre e téren, de a tartalomszolgáltató- és általánosan a szolgáltatóipari rendszereknek szükséges egy utólagos módosíthatóság ahhoz, hogy megbízzanak az adott termékben. A fejlesztésnél ez számításba került, így a rendszer fel van készítve ilyen piaci elvárásokra is.

7.3 További feladatok, bővítési lehetőségek

Bár a rendszer a kezdeti célokat elérte, alkalmas akár valós felhasználók kiszolgálására is. Ettől függetlenül rengeteg feladat maradt, és számos bővítési lehetőség merült fel, melyre a néhány hónapos fejlesztési időszakban nem jutott idő.

- **A rendszer tanításának fejlesztése:** az utószűrésben alkalmazott tartalomtípus alapú tanítás egy merőben új elgondolás, mely a dolgozat egyik legmértékadóbb javaslata. Ennek hatását egy ajánlórendszerre mindenképpen érdemes lenne valós felhasználókon tesztelni.
- **További felhasználói igények felmérése:** a rendszer jelenleg még nem került tesztelésre valós felhasználók által, így lehet arra számítani, hogy a kipróbáláskor sok ötlet, javaslat kerül megfogalmazásra a tesztelők által. Ezen ötletek megfontolása, megtervezése és megvalósítása további feladat.
- **A rendszerterv további fejlesztése, bővítése:** habár az ajánlórendszer alapvető komponensei megtervezésre és kidolgozásra kerültek, további funkciók várhatóak a felhasználói igényektől függően. Ezen funkciókat megfelelően bele kell illeszteni a rendszerbe, majd kidolgozni és lemérni.
- **A rendszer karbantartása:** már a kezdeti prototipikus modellnél is jól láthatóak olyan feladatok, melyekre a rendszer karbantartásakor figyelni kell, azonban a végleges terméknel sok más – előírásból fakadó – követelmény is

elő fog kerülni. Ezeknek a követelményeknek eleget kell tenni, és ehhez a rendszert folyamatosan karban kell tartani, megfelelően kell üzemeltetni.

7.4 Összefoglalás

A modern informatikában megfigyelhető változások radikálisak, gyorsak és nagyhatásúak. A közösségi hálózatok néhány éves elterjedése figyelemre méltó, általuk pedig sok olyan alkalmazási terület nyílt ki, melyek még csak kialakulóban vannak. A közösségi adatok kihasználásában rendkívül sok lehetőség rejlik, így a következő években újabb áttörések várhatóak a területen. Az ajánlórendszerek mostani generációja még csak a kezdet. Kialakulóban vannak nemcsak az e-kereskedelmet, vagy a tartalomszolgáltató rendszereket segítő ajánlórendszerek, de sokkal innovatívabb, teljes szolgáltatásként funkcionáló megoldások is. A jogi és etikai korlátokat átlépve egy ilyen rendszer a jövő informatikájának meghatározó szereplője lehet.

IDÉZETT FORRÁSMUNKÁK

- [1] Ó. Celma és P. Lamere, Szerzők, *Music Recommendation Tutorial*. [Performance]. ISMIR 2009, 2007.
- [2] C. Anderson, Free, 2008.
- [3] C. Anderson, *The Long Tail*, Hyperion, 2006.
- [4] F. Ricci, L. Rokach, B. Shapira és P. B. Kantor, „Introduction to Recommender Systems Handbook,” in *Recommender Systems Handbook*, Springer, 2010, pp. 1-29.
- [5] J. L. Herlocker, J. A. Konstan, L. G. Terveen és J. T. Riedl, „Evaluating Collaborative Filtering Recommender Systems,” 2004.
- [6] G. Adomavicius és A. Tuzhilin, „Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions,” *IEEE Transactions on Knowledge and Data Engineering*, 2005.
- [7] Ó. Celma, *Music Recommendation and Discovery*, Springer, 2010.
- [8] S. P. Harter, „Variations in relevance assessments and the measurement of retrieval effectiveness,” *Journal of the American Society for Information Science*, %1. szám47, pp. 37-49, 1996.
- [9] P. Lops, M. d. Gemmis és G. Semeraro, „Content-based Recommender Systems: State of the Art and Trends,” in *Recommender Systems Handbook*, Springer, 2011, pp. 73-105.
- [10] K. Seyerlehner, „Content-Based Music Recommender Systems: Beyond simple Frame-Level Audio Similarity,” Linz, 2010.
- [11] T. t. N. G. o. R. Systems, „Gediminas Adomavicius; Alexander Tuzhilin;,” *IEEE Transactions on Knowledge and Data Engineering*, %1. szám17, pp. 734-749, 2005.
- [12] Y. Koren és R. Bell, „Advances in Collaborative Filtering,” in *Recommender*

- Systems Handbook*, Springer, 2011, pp. 145-186.
- [13] C. Desrosiers és G. Karypis, „A Comprehensive Survey of Neighborhood-based Recommendation Methods,” in *Recommender Systems Handbook*, Springer, 2011, pp. 107-144;.
- [14] D. F. Bodon, *Adatbányászati algoritmusok*, 2010.
- [15] G. Popescu, „Group recommender systems as a rating problem,” EPFL, 2011.
- [16] C. Lu, J. Lan és Z. Wang, „Aggregation of fuzzy opinions under group decision-making based on similarity and distance,” 2005.
- [17] H.-M. Chu és C.-T. Chen, „Aggregation of fuzzy opinions under group decision making,” in *Fuzzy Sets and Systems*, Elsevier, 1996, pp. 279-285.
- [18] H.-S. Lee, „Optimal consensus of fuzzy opinions under group decision making environment,” in *Fuzzy Sets and Systems*, Elsevier, 2002, pp. 303-315.
- [19] L. Tran és L. Duckstein, „Comparison of fuzzy numbers using a fuzzy distance measure,” *Fuzzy sets and systems*, %1. szám130, pp. 331-341, 2002.
- [20] J. Bu, S. Tan, C. Chen, C. Wang, H. Wu, L. Zhang és X. He, „Music Recommendation by Unified Hypergraph: Combining Social Media Information and Music Content,” in *Proceedings of the international conference on Multimedia*, New York, ACM, 2010, pp. 391-400.
- [21] Zs. Klicsu, „Tartalomszolgáltatáshoz tartozó háttér adatbázis tervezése és implementálása,” Pázmány Péter Katolikus Egyetem Információs Technológiai Kar, 2012.