SZEGEDI TUDOMÁNYEGYETEM
TERMÉSZETTUDOMÁNYI ÉS INFORMATIKAI KAR
MTA-SZTE MESTERSÉGES INTELLIGENCIA KUTATÓCSOPORT

VÉLEMÉNYDETEKCIÓ MAGYAR NYELVŰ SZÖVEGEK ELEMZÉSE ÉS GRÁFANALÍZIS ALAPJÁN

Készítette: Berend Gábor
IV. közgazdasági programozó matematikus

Konzulensek:
Farkas Richárd, tudományos segédmunkatárs
Dr. Csirik János, egyetemi tanár

Szeged, 2008. április

Tartalomjegyzék

Tartalomjegyzék **2**
1. Bevezetés **4**
2. Szövegbányászat **5**
2.1 Véleménydetekció 6
2.2 Online szövegek feldolgozásának nehézségei 7
2.2.1 Általános nehézségek 7
2.2.2 Nyelvspecifikus nehézségek 7
2.3 Kapcsolódó munkák 8
3. Magyar véleménydetekciós korpusz **10**
3.1 A korpusz építésének menete 10
3.2 Annotátorok közötti egyezés (Inter Annotator Agreement) 11
4. A véleménydetektáló rendszer **14**
4.1 Vektortérmodell 14
4.2 Tf-Idf normalizáció 15
4.3 Lemmatizálás 16
4.4 POS-kódok 17
4.5 Tulajdonnevek kezelése 18
4.6 Termek szűrése 18
4.6.1 POS-kód alapú szűrés 18
4.6.2 Stopword alapú szűrés 19
4.6.3 Trigger word alapú szűrés 19
4.7 Döntésifa-alapú osztályozás 20
4.8 1-vs-all szöveg alapú osztályozás 20
4.9 Szövegen kívüli információ beépítése 22
4.9.1 Válaszolási gráf 22
4.9.2 A gráfból kinyert jellemzők ismertetése 24
4.9.3 További jellemzők 24
4.10 A különböző módszerek szavaztatása 25
5. Eredmények **27**
5.1 A döntési fa paraméterei 28
5.2 A különböző egyéni módszerek eredményei 29
5.3 Végső eredmény 30

6. Diszkusszió, további munkák 32
6.1 Diszkusszió 32
6.2 További teendők 33
7. Összefoglalás 35
Bibiográfia 36

1. Bevezetés

Napjainkban a fórumokon, blogokon található folyó szövegekből történő véleménykinyerés a természetesnyelv-feldolgozás egyik intenzíven kutatott feladata, ahol az információkinyeréssel ellentétben nem a dokumentumok tartalma (tények) képezi a vizsgálódás tárgyát, hanem a benne rejlő érzelmi töltet illetve vélemények detektálása. Számos alkalmazása képzelhető el egy véleménydetektáló rendszernek, úgymint fogyasztói elégedettség automatizált felmérése egy termékkel, termékcsoporttal kapcsolatban, vagy például politikai pártok, közszereplők támogatottságának megismerése. Ennek ismeretében például hatékonyabban alakíthatják ki a politikai pártok kampányprogramjukat az embereket leginkább foglalkoztató kérdések ismeretében. Mindehhez meg kell találni a szövegben rejlő úgynevezett pozitív-negatív polaritást, ami nemzetközi szinten is nyitott kérdés, hiszen a természetes nyelv nagyon sokféleképpen képes kifejezni azt.

A munka újszerűsége többek között abban rejlik, hogy ezt megelőzően még nem készült véleménykinyerő rendszer magyar nyelvű szövegekre. A vizsgált szövegek a magyarorszag.hu[1] kormányzati portál fórumának a 2004. december 5-i kettős állampolgárságról szóló ügydöntő népszavazás témájában indított topikjának hozzászólásai voltak. A hozzászólásokat emberi erővel annotáltuk, a hozzászólást író szerzők a témáról alkotott véleményük alapján besorolásra kerültek. Az így elkészült korpusz megteremti a lehetőségét egy olyan statisztikai véleménykinyerő rendszer elkészítésének és empirikus elemzésének, ahol a cél egy-egy hozzászóló polaritásának automatikus meghatározása. Jelen dolgozat tehát tekinthető egy automatikus rendszer elkészítésének megvalósíthatósági tanulmányának. A kidolgozott megközelítések terveink szerint később alkalmazhatóak lesznek más gazdasági ill. politikai témájú véleménykinyerési feladatokra illetve a vizsgált kérdés tárgyalása során felmerülő egyéb aspektusok kinyerésére.

A probléma megoldása folyamán többfajta gépi tanulási módszert alkalmaztunk a szerzőknek véleményük szerinti besorolására. A módszerek egy része a szövegből épített úgynevezett vektortérmodellen alapul. Emellett a feladat megoldása folyamán felhasználtuk az egyes hozzászólásokra érkezett válaszreakciókból épített irányított gráfot is. A végső döntést a különböző megközelítések szavazata hozza meg. Az így kapott eredmények a rendelkezésre álló viszonylag kis tanító adatbázis valamint a magyar nyelv különleges tulajdonságai ellenére (pl. szabad szórend, agglutináció) is csupán néhány százalékkal maradtak el a fórum hozzászólásainak emberi osztályozása során mért annotátorok közötti

[1] http://www.magyarorszag.hu/kapcsolat/parbeszed/agora/nemzet/tema.html?topicid=722

egyetértési szinttől (a feladat bizonytalanságától), mely tekinthető a feladat megoldása során elérhető elvi felső korlátnak, hiszen nem várható, hogy egy számítógépes rendszer az emberi konzisztenciaszintnél pontosabb modellt szolgáltasson.

2. Szövegbányászat

Az elmúlt évtizedekben a számítógépek elterjedése alapjaiban változtatta meg mindennapjainkat életünk szinte minden területén. Az internet térhódítása például megkönnyítette ugyan az információk beszerzését, ugyanakkor idővel újabb problémát generált annak nagy iramú növekedése. Az internet növekedésének magyarázatát Metcalfe hálózatokra vonatkozó törvénye [1, 2] szemlélteti, amely kimondja, hogy „egy hálózat értéke a hozzá csatlakozók számának mértani haladványa szerint növekszik". Egyesek szerint a feltevés téves, és egy hálózat értéke valójában $O(n*\log(n))$ viszonyban áll a hálózatban résztvevők számával [3]. Azt mindenesetre leszögezhetjük, hogy a hálózat értékének növelése szempontjából a hálózathoz csatlakozók számának növelésén keresztül vezet az út.

A méretből eredő probléma lényegét az adja, hogy olyan nagy mennyiségű adat gyülemlett fel időközben, hogy a releváns információk kinyerése, majd pedig ezek feldolgozása csak időigényes és egyúttal költséges emberi erőforrás bevonása árán lehetséges. Az üzleti életben pedig vállalkozások sorsa múlhat a nem megfelelő informáltságon vagy a nem megfelelő időben megszerzett információkon. Mivel az adatok többsége szöveges dokumentumok formájában áll elő, ez a jelenség teremtette meg az igényt az elektronikusan fellelhető, természetes nyelvi szövegek számítógépes feldolgozására (szövegbányászat).

A szövegbányászat az adatbányászat részterületévé vált, amelynek definiálására több megközelítés is létezik [4]:

- „implicit, korábban ismeretlen és potenciálisan hasznos információk adathalmazból történő nemtriviális kinyerése" [5]

-"… automatikus vagy implicit tudást reprezentáló minták kinyerése nagy adatbázisokból, adattárházakból, a webről, … vagy adatfolyamokból" [6]

-"… adathalmazból történő minták felfedezésének folyamata. A folyamatnak automatikusnak (vagy még gyakrabban) félautomatának kell lennie. A megtalált mintáknak hasznosnak kell lenniük… " [7]

-"… rejtett információk megtalálása adatbázisokban" [8]

-"… az egy vagy több gépi tanulási módszer felhasználásával adatbázisból származó adatokban lévő tudásanyag automatikus kinyerésére és analizálására irányuló folyamat." (Roiger, 4. oldal)

Időközben számos más területen, többek között bioinformatikai vagy orvosi projektek keretein belül is sikerrel alkalmazzák a szövegbányászat eredményeit. Dolgozatomban egy manapság nagy ütemben fejlődő részterületre, a véleménydetekcióra (Opinion Mining) fókuszálok.

2.1 Véleménydetekció

Ahogy az internet egyre szélesebb társadalmi csoportokhoz jut el, úgy növekszik a nem csupán tényszerű adatokat közlő, hanem az emberek személyes tapasztalatain alapuló vélemények közlésére szolgáló oldalak – például fórumok vagy blogok – száma és ezáltal a személyes véleményekkel, érzelmi töltettel telített adatok mennyisége is. Az elmúlt években, az internethasználat rohamos növekedésének [9] hatására kellő mennyiségű szöveges dokumentum gyülemlett fel, megnyitva ez által a lehetőséget az emberi vélemények detektálására és összegyűjtésére specializálódott rendszerek elkészítéséhez.

Számos alkalmazása képzelhető el egy véleménydetektáló rendszernek, egy termékekkel kapcsolatos fogyasztói elégedettség automatizált felmérésére irányuló alkalmazástól kezdve, a politikai pártok, közszereplők támogatottságának automatikus felderítésére irányuló rendszereken át.

A véleménykinyerés a természetesnyelv-feldolgozás napjaink egyik nagy erőkkel kutatott szegmense, amely a számítógépes nyelvészet és a szövegbányászat részterülete. A véleményeket tartalmuk szerint legalább két csoportba sorolhatjuk: az elsőbe az egzakt ítéletet hordozó vélemények tartoznak, a másikba pedig a prediktív vélemények. Ítéletet hordozó vélemény például a következő mondat: *„Nagyon tetszett a film!”*, míg a *„Nem hinném, hogy esni fog holnap!”* mondat prediktív vélemény. Az ítéletet hordozó és predikciót tartalmazó vélemények eltérő természetükből adódóan mást is fejeznek ki. Ítéletet hordozó vélemények esetében pozitív-negatív jelentéstartalomról, prediktív vélemények esetében egy esemény bekövetkezésének valószerűségét tartalmazó szövegről beszélhetünk. A kétfajta vélemény gyakorta együttesen is előfordulhat, akár egy mondaton belül is: *„Nagyon élveztem az előadást, bár a többség szerintem unalmasnak találta.”*. Az eltérő típusú vélemények osztályozása eltérő megközelítéseket igényel. [10]

2.2 Online szövegek feldolgozásának nehézségei

Az online szövegek feldolgozása nagy kihívást jelentő feladat. A nehézségeket két fő csoportba oszthatjuk, a feldolgozandó nyelvtől független, általános és a nyelvspecifikus nehézségekre. Ezek taglalására a következő két alfejezetben kerül sor.

2.2.1 Általános nehézségek

Online tartalmak vizsgálata esetében nagy jelentőséggel bír a szöveg előzetes feldolgozása. A weboldalak forrásában először is lokalizálni kell a vizsgálódás szempontjából relevánsnak tartott tartalmat, az adathalmazban lévő zaj minimalizálása érdekében. Hasonló megfontolásból szükségszerű ezek után a meglelt tartalmi részben a formázó információk eltávolítása az oldal struktúrájának figyelembe vétele mellett.

Amennyiben fórumokat vagy blogokat vizsgálunk, úgy érdemes az ott robotok által elhelyezett haszonszerzés vagy weboldalak látogatottságának növelése érdekében elhelyezett kéretlen hirdetések, hozzászólások kiszűrését elvégezni, amelyhez több algoritmus is rendelkezésünkre áll. [11]

Szintén fórumok és blogok esetében a szokottnál gyakoribbak lehetnek az elírások, hangulatjelek és speciális nyelvezet használata, amelyeknek az alkalmazás típusától függően nem árt figyelmet szentelni.

Megemlíthető ezek mellett még a homoníma (azonos alakúság) és poliszémia (többértelműség) viszonylag gyakori előfordulása a szövegben. Ezek feloldása sokszor az emberi agy számára is nehézkes, a számítógépnek azonban még az egyszerűbb esetek egyértelműsítése sem feltétlen megoldható, amennyiben a gépnek nincsenek ismeretei a világ alapvető törvényeiről. Az *„Öt holdja van.”* mondat csillagászati kontextusban csak azt jelentheti, hogy egy bolygó körül 5 égitest kering, míg mezőgazdasági kontextusban azt takarja, hogy valakinek például 5 katasztrális holdnyi (=8000 négyszögölnyi) földje van.

2.2.2 Nyelvspecifikus nehézségek

A szövegekben rejlő, úgynevezett pozitív-negatív polaritás megtalálása nemzetközi szinten is nyitott kérdés, hiszen a természetes nyelv nagyon sokféleképpen képes kifejezni azt. Ez az állítás azonban a magyar nyelv esetében hatványozottan helytálló. Ennek okát számos jelenségben kereshetjük.

A magyar nyelv agglutináló, ami azt jelenti, hogy a szavak jelentését elsősorban a szótári szóalakok (lexémák) megváltoztatásával állítja elő, toldalékok hozzákapcsolásával. Részben az agglutináló nyelvekre jellemző strukturális, szintaktikai sajátosságok nehezítik meg a feldolgozást, ami következtében egyazon lexéma számos alakot ölthet. A *„Nem tetszett a film!”* és a *„Nem tetszik ez a film!”* mondatok bár ugyanazt a véleményt fejezik ki, a *tetszik* ige mégis két eltérő formában fordul elő bennük. Ennek kiküszöbölésében nyújt segítséget a későbbiekben ismertetésre kerülő szótövesítés (lemmatizálás) módszere.

Nehézséget okoz még a szabad szórend használatának lehetősége, amivel az indoeurópai nyelvek esetében nem találkozunk, és ami jelentősen megnöveli az azonos jelentéstartalmak kifejezésének lehetőségeinek számát. Sőt, eltérő szórend használatával lehetőség van eltérő tartalmak kifejezésére is, amelynek detektálása szintén nehézségeket okoz.

Végső soron pedig a magyar web méretéből adódóan egyes domainekre nehezebb megfelelő méretű szöveges adatbázist építeni a weboldalak tartalmából, hiszen az interneten fellelhető mintegy 11,5 milliárdra becsült publikusan indexelt weboldalnak [12] a magyar nyelvűek csupán a 0,1%-át teszik ki[2].

2.3 Kapcsolódó munkák

A véleménydetekció csak az elmúlt 3-4 évben került a tudományos érdeklődés középpontjába, azonban egyértelmű növekedés figyelhető meg az e témában megjelent publikációk számában.

Kim és Hovy 2007-es cikkükben [10] a jövőre vonatkozó, prediktív vélemények kinyerését célozták meg. A munkájukban bevezetésre került, *Crystal* névre hallgató rendszer a kanadai választások eredményének fórumozók szerinti végkimenetelére tesz predikciót. Jelen dolgozat témája ehhez hasonló, mivel ez is egy szavazásról szóló fórumot vizsgál, csak magyar nyelven. A két munka döntően abban különbözik, hogy míg [10] a választási eredményekre irányuló prediktív véleményeket nyer ki, addig munkámban a hozzászólók népszavazáson tanúsított magatartását próbáltam meghatározni (ítéletet hordozó vélemények alapján).

Kobayashi és szerzőtársai 2007-ben megjelent munkájukban [13] a vélemények tárgyának és pozitív-negatív tartalmának együttes meghatározását tűzték ki célul. A megoldás

[2] Forrás: http://www.rangsorolo.hu/help.jsp?chapter=klaszter.jsp

során először 116 éttermekkel foglalkozó japán blogot vizsgáltak, majd ezt bővítették ki még telekommunikációval, valamint az autó- és videojáték-iparral foglalkozókkal.

A pozitív-negatív polaritáson túl, a szubjektivitás-objektivitás detektálása adja egy másik aspektusát a vizsgálódásoknak. Esuli és Sebastiani 2006-os publikációjában [14] vezette be a WordNet ontológia analógiájára létrehozott SentiWordNetet, amiben minden egyes synsethez definiálták annak objektivitását, pozitív illetve negatív irányultságát számszerűsítő számhármasait.

Kaji és Kitsuregawa az internet segítségével épített pozitív-negatív szavakat és kifejezéseket tartalmazó tezauruszt.[15] Kutatásuk lényege abban állt, hogy nagy mennyiségű japán nyelvű HTML dokumentumokból, strukturális jegyek alapján csak a nagyon magas pontosságot mutató pozitív-negatív orientáltságú kifejezéseket, szavakat nyertek ki.

3. Magyar véleménydetekciós korpusz

3.1 A korpusz építésének menete

Tudomásunk szerint korábban nem létezett magyar nyelvű véleménykinyerési kutatásokhoz felhasználható, szakértői jelöléseket tartalmazó korpusz. Munkánkhoz tehát saját korpuszt építettünk. A vizsgált szövegeket a magyarorszag.hu kormányzati portál fórumának a 2004. december 5-i kettős állampolgárságról szóló ügydöntő népszavazás témájában indított topikjának [16] hozzászólásai képezték. A hozzászólások elemzése során a kampánycsendet megelőző 3 hónapban elküldött hozzászólások lettek feldolgozva.

A kapott 1294 hozzászólás két nyelvész segítségével került annotálásra (akik a munkát egymástól függetlenül végezték, lehetőséget teremtve az annotáció konzisztenciaszintjének mérésére). Az annotátorok azt a feladatot kapták, hogy az egyes hozzászólások szövegei alapján (azaz nem vehettek figyelembe korábbi hozzászólásokat illetve nem ismerték a hozzászóló személyét), amennyiben egyértelműen eldönthető, hogy az illető IGEN-nel vagy NEM-el szavazna a népszavazáson, azt jelöljék. Az annotálási munkát egy, korábban jelentés-egyértelműsítésre fejlesztett annotációs szoftver segítette.

A tapasztalatok azt mutatták, hogy a kettő élesen elkülöníthető alaposztályon túl fel kell venni egy ÉRVÉNYTELEN és egy SEMMI címkét is az üzenetek osztályozására. IGEN illetve NEM címkét rendre azok a hozzászólások kaptak, amelyeket elolvasva eldönthető volt, hogy szerzője részt kíván venni a népszavazáson, és igennel illetve nemmel fog szavazni. ÉRVÉNYTELEN besorolást kapott mindazon hozzászólás, amely tartalmából kiderült, hogy szerzője vagy nem vesz részt a népszavazáson, vagy amennyiben részt vesz, úgy a feltett kérdésre szándékosan igen és nem választ is ad egyidejűleg, érvénytelen szavazatával is protestálva a szavazásban feltett kérdés tartalma ellen. Végül a SEMMI kategóriába azok a hozzászólások kerültek, amelyek nem voltak besorolhatók az előbbi három osztály egyikébe sem, azaz nem tartalmaztak információt a kettős állampolgárságról szóló népszavazás kérdésével kapcsolatban.

Ezt követően, az egyes hozzászólások címkéjének ismeretében a szerzők automatizált osztályozása is megtörtént. Szerzők csak akkor kerültek SEMMI besorolásba, ha kizárólag SEMMI-típusú hozzászólások származtak tőlük a vizsgált időszakban. Minden egyéb esetben a hozzászólók abba az osztályba kerültek, amilyen tartalmú, nem SEMMI-nek minősített

hozzászólásukból a legtöbbjük volt. Esetleges egyezés esetén az időben utolsónak elküldött üzenetének javára lett meghatározva a címke.

A különböző osztályok közötti eloszlásokat az alábbi táblázat tartalmazza, a hozzászólások valamint a hozzászólók szintjén:

	Hozzászólások		**Hozzászólók**	
	(db)	(%)	(db)	(%)
IGEN	186	14,374	22	25,882
NEM	160	12,365	29	34,118
ÉRVÉNYTELEN	26	2,009	7	8,235
SEMMI	922	71,252	27	31,765
Összesen	1294	100	85	100

3.1.1. táblázat – A hozzászólások és hozzászólók egyértelműsítés utáni címkéinek eloszlása

A táblázatra pillantva megállapíthatjuk, hogy a nem kategorizálható SEMMI címkével ellátott hozzászólások aránya igen magas volt a hozzászólásokon belül. Az általuk okozott zaj minimalizálására tett kísérleteket a következő fejezet tartalmazza.

3.2 Annotátorok közötti egyezés (Inter Annotator Agreement)

Az annotálás során tapasztalt 299 darab eltérő osztályba sorolt hozzászólás végleges címkéjének meghatározását egy harmadik, független nyelvész végezte. Az annotátorok döntése az esetek 76,89%-ban, 995 alkalommal egyezett meg a hozzászólások szintjén, és a közöttük fennálló egyetértési szintet számszerűsítő, ún. κ-mérték értéke 0,598947 volt.

A κ- mértéket kiszámítására használt képlet a

$$\kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)}$$

3.2.1. képlet – a κ-mérték kiszámítására alkalmazott képlet

,amelyben Pr(a) az annotátorok között megfigyelt egyezés relatív gyakorisága, Pr(e) pedig a véletlen által bekövetkező egyezések valószínűsége. A κ-mérték ereje abban rejlik, hogy az osztályozás jóságának számítása során eliminálásra kerül a véletlennek köszönhető

azonos jelölések hatása, így jobb mérőszámot ad a jelölések hasonlóságának eldöntésére. A κ-mérték általánosan elfogadott interpretációja az alábbi táblázatban található:

κ –mérték	**Interpretációja**
= 0,00	Nincs egyezés
0,01 - 0-20	Nagyon alacsony egyezés
0,21 - 0,40	Alacsony egyezés
0,41 - 0,60	Közepes egyezés
0,61 - 0,80	Nagyfokú egyezés
0,81 - 0,99	Közel tökéletes egyezés
= 1,00	Tökéletes egyezés

3.2.2. táblázat – a κ-mérték interpretációja

A két annotátor hozzászólás szintjén tett jelöléseinek összegzése az alábbi, ún. konfúziós mátrixokban találhatók:

		Annotátor A				
		IGEN	NEM	ÉRVÉNYTELEN	SEMMI	**Összesen**
Annotátor B	IGEN	**106** 8,19%	**3** 2,32%	**1** 0,08%	**125** 9,66%	**235** 18,16%
	NEM	**5** 0,39%	**83** 6,41%	**1** 0,08%	**106** 8,19%	**195** 15,07%
	ÉRVÉNYTELEN	**-**	**-**	**16** 1,24%	**2** 0,15%	**18** 1,39%
	SEMMI	**22** 1,70%	**29** 2,24%	**5** 0,38%	**790** 61,05%	**846** 65,38%
	Összesen	**133** 10,23%	**115** 8,89%	**23** 1,78%	**1023** 79,06%	**1294** 100,00%

3.2.3. táblázat – Az annotátorok hozzászólás szintű konfúziós mátrixa

Ezzel szemben, ha a hozzászólások végleges címkéje ismeretében, a már ismertetett módon a hozzászólások íróit osztályoztuk, 0,653220-es κ-mérték volt megfigyelhető. Ekkor a 85 hozzászólóból 62 került azonos besorolásba a két annotátor elsődleges jelölése szerint, ami

72,941%-os egyezést jelent. A döntési modell építése alatt ezt a pontossági szintet sikerült megközelíteni, és csupán néhány százalékkal elmaradni tőle.

Az annotátorok hozzászólók szintjén tett előzetes jelöléseinek eloszlása az alábbi táblázatból olvashatók ki:

		Annotátor A				
Annotátor B		IGEN	NEM	ÉRVÉNYTELEN	SEMMI	**Összesen**
	IGEN	16 18,82%	2 2,35%	2 2,35%	6 7,06%	26 30,56%
	NEM	3 3,53%	20 23,53%	-	4 4,71%	27 31,76%
	ÉRVÉNYTELEN	-	-	4 4,71%	-	4 4,71%
	SEMMI	2 2,35%	4 4,71%	-	22 25,88%	28 32,94%
	Összesen	21 24,71%	26 30,59%	6 7,06%	32 37,65%	85 100,00%

3.2.4. táblázat – Az annotátorok hozzászóló szintű konfúziós mátrixa

4. A véleménydetektáló rendszer

4.1 Vektortérmodell

Szöveges tartalmak tömör reprezentációjára a vektortérmodell (VTM) nyújtja a legszélesebb körben használt megoldást a szövegbányászatban. A modell minden egyes dokumentumot egy vektorral ír le, amelyben minden elem az egyes termek (általában szavak) előfordulását jelenti. Termek alatt a reprezentáció egységeit, alapesetben az írásjelek által határolt szavakat (unigram) értjük. Bizonyos módszerek több szóból álló kifejezéseket (n-gram) is alkalmaznak reprezentációs egységként, amellyel általánosságban jobban jellemezhetők a dokumentumok, azonban ez a fajta reprezentáció jelentősen megnöveli a dokumentumok feldolgozásának (indexelésének) idő- és tárigényét. A kutatás során különböző módon szűrt uni- és bigramok alapján végeztem méréseket.

A dokumentumhalmazhoz tartozó lexikon mindazon termeket tartalmazza, amelyek legalább egy alkalommal, legalább egy dokumentumban előfordultak a vizsgált korpuszban, a dokumentumokat leíró vektorok dimenziószáma pedig a lexikon méretével egyezik meg. Az egyes dokumentumok közötti hasonlóságok mértékének meghatározása a termek által feszített vektortérben történik meg.

A vektortérmodellben az adathalmazt egy T^{NxM} term-dokumentum előfordulási mátrixszal reprezentálhatjuk, ahol N-el jelöljük a dokumentumokat megtestesítő vektorok dimenziószámát, M-el pedig a dokumentumok számát, és tetszőleges t_{ij} eleme az i-edik token j-edik dokumentumban való előfordulásait szimbolizálja.

Egy token j-edik dokumentumra értett *„fontosságát”* több módon is tárolhatjuk. A legegyszerűbb választás, ha binárisan kezeljük a szavak előfordulását, és a mátrix t_{ij} elemének értéke 1, ha az i-edik token előfordul a j-edik dokumentumban, máskülönben 0. Más megközelítésben t_{ij} tartalma lehet az i-edik token j-edik dokumentumban lévő összes előfordulásának száma is. A termek súlyainak meghatározása során az általánosan használt megoldás a tf-idf indexek használata, amivel a következő fejezet foglalkozik részletesebben.

A vektortérmodellről általánosságban elmondható, hogy egyszerű, gyors modell, melyet használva könnyűszerrel számolhatunk dokumentumok közti távolságokat, vagy alkalmazhatunk ebben a térben standard gépi tanulási technikákat. Az is igaz azonban, hogy a modell a termek előfordulását függetlennek tekinti, ami a valóságban nem helytálló feltételezés, valamint nem kezeli a termek közti szintaktikai kapcsolatokat, szemantikai

tartalmat. A reprezentált tér dimenziószáma általában ráadásul magas, a term-dokumentum előfordulási mátrix pedig igen ritka, hiszen egy dokumentumban általában a teljes lexikon termjeinek csak egy elhanyagolható hányada szerepel, ami körülményessé teszi alkalmazását.

4.2 Tf-Idf normalizáció

A tf-idf (term frequency–inverse document frequency) értékek kiszámítása jó szolgálatot tesz a teljes lexikon legrelevánsabb részhalmazának megállapításában, ami egyúttal a dokumentumok feldolgozásának idő-és tárigényének csökkenésére is fölhasználható, hiszen a szótárban szereplő termek halmazát leszűkíthetjük egy alkalmasan megválasztott küszöbszám feletti érétkűekre. Lényege abban rejlik, hogy a termeket úgy súlyozza, hogy azokat látja el magas értékkel, amelyek előfordulása egy dokumentumban a leginkább jellemzi a dokumentum tartalmát.

Képlete a

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

4.2.1. képlet – i-edik term j-edik dokumentumra vonatkozó tf értékét kiszámító képlet

módon számított, i-edik term j-edik dokumentumban való előfordulásának teljes gyakoriságát tartalmazó term frekvencia, valamint az

$$idf_i = \log \frac{|D|}{|\{d_j : t_i \in d_j\}|},$$

4.2.2. képlet – i-edik term idf értékét kiszámító képlet

, ahol a hányados számlálójában szereplő szám a dokumentumok számát, a nevező pedig az i-edik termet tartalmazó dokumentumok számát tartalmazó invertált dokumentum gyakoriságok szorzataként

$$tfidf_{i,j} = tf_{i,j} * idf_i$$

4.2.3. képlet –tf-idf értékének kiszámításának képlete

formában áll elő, biztosítva ezáltal, hogy azok a dokumentumok érjenek el magas értéket egy termre nézve, amelyekre igaz, hogy a term hangsúlyos az adott dokumentumon belül, míg más témájú dokumentumokban nem jellemzően szerepelnek. Nyilvánvaló, hogy a vizsgált domainen a *„választás”* szó jelenléte vagy éppen hiánya prediktívebb tulajdonság egy hozzászólás tartalmára nézve, mint például a szinte minden üzenetben felfedezhető kötőszavak, névelők vagy éppen határozószók.

A tf-idf értékének ez a tulajdonsága teszi az információ-visszakeresés hasznos eszközévé. A korábban bevezetett T^{NxM} term-dokumentum előfordulási mátrixban t_{ij} értéke tfidf(d_j, t_i)-vel lesz egyenlő, ahol tfidf(d_j, t_i) az i-edik term j-edik dokumentumra számított tf-idf értéke. Ezen értékeknek kiszámítására a JAVA nyelven implementált, nyílt forráskódú Text Clustering Toolkit-et (TCT)[3] alkalmaztam.

4.3 Lemmatizálás

A tf-idf érték alapján történő term-szűrésen túl a lemmatizálás is segíthet a termek dokumentumokban való előfordulásának reprezentálása során jelentkező zaj valamint a felhasználásra kerülő tárhely csökkentésében. Célja, hogy az eltérő ragozásban álló lexémák szótári alakjuk szerinti kanonizált formában, egy termként jelentkeznek, így nem okoz gondot, ha ugyanaz a jelentéstartalom eltérő nyelvtani szerkezetben (eltérő személyben) fordul elő két dokumentumban. Ez a tulajdonsága főleg a magyarhoz hasonló agglutináló (toldalékoló) nyelvek esetében nyújt hasznos segítséget a folyó szövegek feldolgozásában, ahol egy lexéma számos formában fordulhat elő [17].

Ugyan a természetes nyelvfeldolgozás egy hasznos eszköze a lemmatizálás, óvatosan kell bánni vele, és figyelni kell a tokenek ún. túlgeneralizálására, hiszen a toldalékok leválasztásával egyes esetekben értékes jelentés-megkülönböztető szereppel bíró toldalékokat távolíthatunk el. Erről a jelenségről számol be [18], amely szerint gyenge visszaesés volt tapasztalható alkalmazásuk eredményeiben, amennyiben a lemmalizált szövegen végezték a gépi tanulást. Ennek egyik oka például az volt, hogy a negatív vélemények megfigyelhetően nagyobb számban szerepeltek múlt időben, amit a lemmatizálás után már nem tudtak fölhasználni.

[3] http://mlg.ucd.ie/content/view/18/

4.4 POS-kódok

A lemmatizálás elvégzése előtt mód nyílik a ragozott szóalakokon a morfológiai kódok segítségével azok POS-kódjának meghatározására. POS (Part Of Speech)–kódok alatt a termek szófaji kódját értjük. Meghatározásukkal hasznos információkkal gyarapszunk, amit a vektortérmodellünk építése során sikeresen kamatoztathatunk. A POS-kódok figyelembe vételével lehetővé válik az azonos alakú, ám a mondatban eltérő funkciókat betöltő, és ez által eltérő tartalmat kifejező lexémák eltérő módon történő kezelése. Esetünkben hasznos volt például a *„nem”* szó főnévként („NEM-el fogok szavazni”) és a határozószók („én nem úgy gondoltam”) egyik vállfajaként, módosítószóként való megkülönböztetése. Ezzel természetesen növekszik vektortér dimenziószáma, de a jelenség ellensúlyozható, többek között a már eddig ismertetett és a későbbiekben ismertetésre kerülő módszerek valamelyikével.

A POS-kódok meghatározása az egyes mondatokban előforduló termek potenciáls szófaji kódjainak valószínűsége alapján történik. A lemmatizáláshoz valamint a szófaji kódok megállapításához egyaránt egy, a SZTE Mesterséges Intelligencia Kutatócsoportja által magyar nyelvre fejlesztett szoftvert használtam [19]. Az alábbi táblázat a POS–kódok fő osztályait tartalmazzák:

POS osztályok	**Jelölésük**
1. Főnevek (Noun)	N
2. Igék (Verb)	V
3. Melléknevek (Adjective)	A
4. Névmások (Pronoun)	P
5. Névelők (Article)	T
6. Határozószók (Adverb)	R
7. Névutók (Adposition)	S
8. Kötőszók (Conjunction)	C
9. Számnevek (Numeral)	M
10. Mondat értékű szavak (Interjection)	I
11. Egyéb (Residual)	X
12. Rövidítések (Abbreviation)	Y
13. Nyílt tokenosztályok (Open/Other)	O

4.4.1. táblázat –POS-kódok fő csoportjainak táblázata

A fenti táblázat osztályai további, itt részletezésre nem kerülő alosztályokra bonthatók, pontosabb képet nyújtva ez által arról, hogy egy-egy szó milyen ragozásban áll a mondatban.

4.5 Tulajdonnevek kezelése

Általánosságban elmondható, hogy a tulajdonneveket vagy rövidítéseket tartalmazó mondatok nagyobb valószínűséggel bírnak releváns információtartalommal, mint az azokat nem tartalmazó társaik, ezért is érdemes külön foglalkozni ezen mondatok felismerésével, illetve a tulajdonnevek lokalizálásával a mondaton belül.

Minden nem mondat elején szereplő nagy kezdőbetűs szó vagy szókapcsolat esetén azzal a feltételezéssel éltünk, hogy az egy tulajdonnév, ami legközelebbi előfordulásakor már akkor is tulajdonnév besorolásba került, ha az a mondat elején szerepelt. Ezeket a token sorozatokat rendre a NAMED_ENTITY termre cseréltük le.

Ennek a rendszerre nézve az a jelentősége, hogy többek között az olyan megszólítások, mint például a *„Kedves Sándor!”* vagy a *„Kedves Mária!”* azonos elbírálás alá estek. Ezt ezúttal minden további nélkül megtehettük, hiszen a hozzászólásokat az annotáláskor is függetlenül kezeltük, vagyis nem vettük figyelembe, hogy kit szólítanak meg, illetve kinek a hozzászólására reagálnak bennük.

4.6 Termek szűrése

A tf-idf szűrések mellett más módon is lehetőség van akár a termek, akár pedig a vizsgált dokumentumok megszűrésére. A következő alfejezetek ezeket a részben problémaspecifikus módszereket veszik sorra, és hasonlítják össze a tf-idf szűréssel.

4.6.1 POS-kód alapú szűrés

A termek POS-kódját figyelembe vevő szűrésekor csak azokat a termeket vesszük be a vektortérmodell építésébe, amelyek a POS-kódok egy előzetesen meghatározott részhalmazából valók. Korpuszról korpuszra változik, hogy mely szófajú termek megtartása mellett érhető el a legjobb eredmény. Nagy adatbázis esetén nem feltétlen szükséges mind a 2^{14}-1 szóba jöhető, lehetséges részhalmaz meghatározása, hanem helyette mohó algoritmust is segítségül hívhatunk egy megközelítőleg optimális részhalmaz kialakításához.

Azon szófajok esetében, amelyek elemei zárt halmazt alkotnak, és tudjuk róluk, hogy tartalmat nem fejeznek ki, mint például a névelők, megtehetjük, hogy egyáltalán nem vizsgáljuk őket. Ezzel egyrészt csökkenteni tudjuk a POS-kódok optimális részhalmazának megtalálásához szükséges 2^{14}-1 számú kiválasztást, másrészt olyan szavak vizsgálatától tekintünk el, amelyek a tf-idf szűréssel egyébként is kiesnének. Az előzetes szűrés tehát idő-és tárhelycsökkentő tulajdonságokkal is bír.

A bemutatott feladat megoldása során az a szófaji szűrés bizonyult a legjobbnak, amely a főneveket, igéket, mellékneveket és határozószókat vette csak figyelembe.

4.6.2 Stopword alapú szűrés

Egy nyelv *stopword*-jei azon szavak listája, amelyek nem bírnak semmiféle jelentéstartalommal vagy szemantikai relevanciával. Az interneten fellelhető ezeknek a szavaknak az általánosan elfogadott listája az egyes nyelvekre. [20]-ban például 23 nyelv stopwordjei szerepelnek.

A listákban szereplő szavak többsége ugyan a tf-idf szűrés során is eliminálásra kerülne, de mivel nincs olyan domain, ahol jelenlétük jelentésmegkülönböztető szereppel bírna, így minden további nélkül elvégezhetjük előzetes eltávolításukat, ez által is tovább csökkentve alkalmazásunk műveleti - és tárhelyigényét.

4.6.3 Trigger word alapú szűrés

Trigger word alatt azokat a szavakat értjük, amelyek jelenléte biztosítja, hogy releváns dokumentummal (mondattal) van dolgunk. Ezek listái a stopwordökével ellentétben nem állandó érvényűek, tartalmuk feladatról feladatra változik.

A kettős állampolgárság kontextusában intuitíve az *„igen”*, *„nem”* és a *„támogat”* lexémákat találtam ilyen szavaknak. A szűrést két módon is végrehajtottam. Egyik esetben a szűrést hozzászólás szinten végeztem el, vagyis csak azok a hozzászólások kerültek vizsgálatra, amelyek a fenti trigger word-ök közül legalább egyet tartalmaztak. Az ennél szigorúbb szűrés mondat szinten lett végrehajtva, vagyis az előző szűrés után megmaradt hozzászólások mondatai közül is csak azok kerültek bele a modellbe, amelyekben szerepelt valamelyik trigger word. Ezeknek a szűréseknek a hátterében az a megfontolás állt, hogy a megszűrt hozzászólások (mondatok) szinte biztosan a népszavazáshoz kapcsolódó tartalmat

fejeznek ki, azaz csökkenthető a viszonylag nagy arányú, nem a hozzászóló szavazási szándékát reprezentáló mondatokból eredő zaj mértéke.

4.7 Döntésifa-alapú osztályozás

A VTM-beli tanulás elvégzésére döntési fát alkalmaztam. [21] Használata azért is volt előnyös esetünkben, mert diszkrét jellemzőkre kifejezetten jól alkalmazható, valamint abból kifolyólag, hogy mohó módon határozza meg a döntési csomópontokat, nagy dimenziós terekben is hatékonyan működik. Outputja mindemellett az ember számára is könnyedén értelmezhető.

Az ID3 döntési fán alapuló tanulóalgoritmus egyik változata a C4.5 [22], amellyel előre definiált, diszkrét osztályok felügyelt tanulását végezhetjük el. A tanulási folyamat outputja egy tengelypárhuzamos vágásokat alkalmazó döntési fa. Ez azt jelenti, hogy a tanulás során az ismert címkéjű entitások alkotta teret részterekre osztjuk tengellyel párhuzamos hipersíkok mentén. Így hierarchikusan rendezett, és ezáltal fa formára hozható, címkével ellátott, d-dimenziós alakzatot kapunk a térben. Mivel a C4.5 a d-dimenziós tér pontjaiként kezeli a tulajdonságvektorokat, folytonos értékekkel jellemezhető tulajdonságok alapján is elképzelhető a tanulás. A C4.5 tudásreprezentációra az ún. *„oszd meg, és uralkodj”* elvet érvényesíti, amely értelmében a tanulás folyamán egy résztér abban az esetben kerül fölbontásra, ha a benne található elemek nem kellőképpen homogének. Mivel a felbontás tengelypárhuzamos hipersíkokkal történik, így a tanulás nagyon gyors. A módszer egyik fő előnye annak műveleti igényében rejlik, az ugyanis legrosszabb esetben is $O(dn^2)$, ahol d a tulajdonságok száma, n pedig a tanuláshoz felhasznált minták száma.

A tanulásokat az új-zélandi Waikato Egyetemen kifejlesztett, JAVA-ban íródott, szabad forráskódú, adatbányászati alkalmazásokat támogató Weka [23] szoftverrel hajtottuk végre, és a C4.5-ös döntési modell Wekában implementált megfelelőjét a J48-as tanulóalgoritmust használtuk.

4.8 1-vs-all szöveg alapú osztályozás

Az általános 4-osztályos VTM-alapú tanulás eredményeinek vizsgálata során arra a következtetésre jutottunk, hogy a SEMMI osztály annyira diverz, hogy nem lehet annak hatékony direkt osztályozását megtanulni. Ez a megfigyelés nem meglepő, hiszen a SEMMI osztály a teljes, nem a témához tartozó univerzumot reprezentálja, amelyre lehetetlen kis

számú, nem témába illő hozzászólás alapján értelmes modellt alkotni. A tanulási feladatot eszerint átfogalmaztam, úgy, hogy a cél nem egy 4-osztályos modell tanulása, hanem az IGEN, NEM és ÉRVÉNYTELEN osztályok megtanulása legyen. Ebben az esetben, minden ezekbe nem besorolható elem kapja végül a SEMMI címkét. A három „tanulható" osztályt az összes többi osztállyal egyenként szembeállítottam, és az optimalizálás tárgyát a vizsgált osztályra vonatkozó feltételes osztályvalószínűségek maximalizálása képezte. Az imént bevezetett feltételes valószínűséget tetszőleges j-edik hozzászólóra és i-edik osztálycímkére a

$$P(classlabel_i \mid term) = \frac{\sum |\{d : term \in d \wedge d \in D_{classlabeli\,i}\}|}{\sum |\{d : term \in d \wedge d \in D\}|}$$

4.8.1. képlet – A feltételes osztályvalószínűség

képlettel számítottam ki, amely számlálója az i-edik típusú osztályba tartozó fórumozók adott *term*-et tartalmazó hozzászólásainak halmazának elemszámával egyenlő, nevezőjében pedig a teljes dokumentumhalmaz *term*-et tartalmazó hozzászólásainak részhalmazának elemszáma található.

Ennek a feltételes valószínűségnek az értéke csak abban az esetben 1,0, ha a teljes dokumentumhalmazban egy termet csupán egyféle osztálycímkével ellátott szerzők használtak, így az az azzal a címkével rendelkező fórumozók egy jellemző szava.

Minden címkére, minden egyes termre meghatározhatók ezek a feltételes valószínűségek, az elgondolás pedig az volt, hogy csupán az egy bizonyos küszöbszám feletti értékű termek jelenlétének vizsgálata alapján határozzuk meg egy hozzászóló címkéjét. A mérési eredmények azonban azt mutatták, hogy az adatok akkora zajjal vannak terhelve, hogy csupán az 1,0 feltételes valószínűséggel rendelkező szavakat érdemes vizsgálni. Így minden *c* osztálycímkére elkészíthetők a

$$W_{c,i} = \{t : P(c_i \mid t) = 1{,}0\}$$

4.8.2. képlet – az 1,0-ás feltételes c-osztályvalószínűségű termek halmaza

diszjunkt halmazok, amelyek a termeknek csupán azt a részhalmazát tartalmazták, melyeknek a 4.8.1. képletben bevezetett feltételes osztályvalószínűsége 1,0.

Egy hozzászóló végleges *l* címkéje végül az

$$l = \arg\max_{c_i \in classlabel} \frac{\sum |\{t : t \in \{D_j \cap W_{c,i}\}\}|}{|W_{c,i}|}$$

4.8.3. képlet – A hozzászólók címkéjének meghatározása

képlet segítségével határozható meg, ahol $W_{c,i}$ a 4.8.2. képletnek megfelelő halmazt, D_j pedig a j-edik hozzászóló hozzászólásainak termjeinek halmazát jelöli.

4.9 Szövegen kívüli információ beépítése

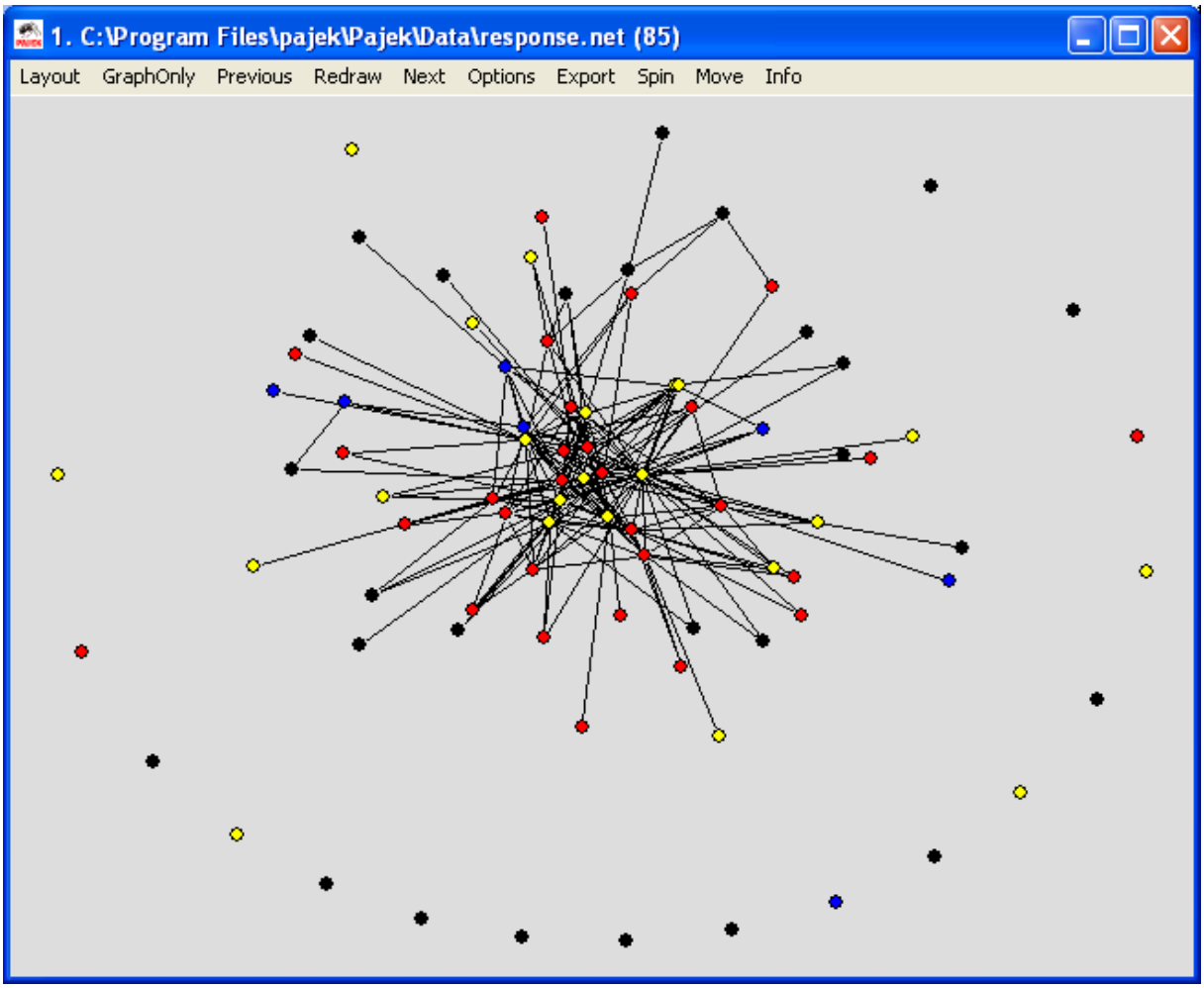
A hozzászólások szövegeiben rejlő tartalmi információn túl más jellegű, a fórum HTML forrásából kinyerhető információkat is felhasználtunk az osztályozás során.

4.9.1 Válaszolási gráf

A fórumozók kategorizálásához hasznos információt tartalmazhat a hozzászólók egymás közötti viszonya, amelybe az egymásnak adott válaszok adhatnak némi betekintést. Hipotézisünk az volt, hogy a nem SEMMI osztálycímkével ellátott hozzászólók közötti üzenetváltások vannak túlsúlyban, valamint meg kívántuk vizsgálni, hogy az eltérő álláspontot képviselő fórumozók gyakrabban válaszolnak-e egymásnak, mint az azonos véleményen lévők.

A hozzászólók közötti kapcsolatok analizálására elkészítettük a G = (V,E) súlyozott, irányított gráfot, ahol a csúcsok az egyes hozzászólókat reprezentálják, *a* és *b* csúcsok között pedig akkor, és csak akkor megy él, ha a gráfban *a* pontként szereplő hozzászóló valamely hozzászólására a *b* pont által reprezentált hozzászóló valamikor választ intézett, súlya pedig az ilyen hozzászólások számával egyezik meg. Ezek az információk a fórum HTML-forrásának vizsgálatával megállapíthatók, ugyanis az üzenetküldőknek lehetőségük van feltüntetni, mely másik fórumozó hozzászólásra kívánnak reagálni. Azt azonban érdemes megjegyezni, hogy ezzel a lehetőséggel nem mindenki él, és akadnak olyan fórumozók is, akik ezt a funkciót nem ilyen értelemben használják, hanem például saját hozzászólásukat „kommentálják”. Ezeket, a gráfban hurokélként jelentkező éleket nem vettük figyelembe a gráf vizsgálata során.

A válaszolási gráf vizualizációjára az ingyenesen hozzáférhető, hálózatok analizálásra készített Pajek[4] [24] szoftvert használtuk. A hozzászólók közötti kapcsolatok szemléletessé tételét az alábbi ábra szolgáltatja:



4.9.1. ábra – A válaszolási gráf egy vizualizációja

A pirosra, citromsárgára, kékre, illetve feketére színezett csúcsok rendre az NEM, IGEN, ÉRVÉNYTELEN illetve a SEMMI osztálycímkével rendelkező hozzászólókat reprezentálják. Már az ábra alapján is látható, hogy egy ilyen gráf elkészítése milyen hasznos is lehet, hiszen egyértelműen megállapítható, hogy a leginkább megszólított emberek hozzászólásainak többsége érdemi tartalommal bírt a vizsgálat szempontjából. A további hasznos jellemzők ismertetése a következő alfejezet feladata lesz.

[4] http://pajek.imfm.si/doku.php?id=pajek

4.9.2 A gráfból kinyert jellemzők ismertetése

A gráfból az egyes csúcspontokra vonatkozólag kigyűjtöttünk számos jellemzőt és a gépi tanuló algoritmusokra bíztuk a releváns összefüggések megtanulását. A válaszolási gráf alapján épített tulajdonságvektorok tartalmazták a hozzászólóknak megfeleltetett csúcsok ki- és befokait, valamint az ezen értékek dekompozíciójából származó IGEN-es és NEM-es címkével ellátott ki- és befokokat, illetőleg ezek arányait a csúcs összes ki- és befokának viszonyában. IGEN (NEM) címkéjű befokok (kifokok) alatt az IGEN (NEM) címkéjű szomszédból (szomszédba) vezető élek számát értjük.

A közvetlen szomszédok vizsgálatán túl, meghatározásra kerültek a különböző módokon, 2-hosszú úton elérhető csúcsok számai aggregálva, valamint IGEN-es és NEM-es címkéjű élekre lebontva egyaránt. Ennek az információnak az ismeretében megkülönböztethető egy olyan fórumozó, aki csak egy-egy izolált (esetleg nem is a témához szorosan kapcsolódó) esetben keveredett vitába valakivel azok közül, akik tendenciózusan sokakat késztettek válaszreakcióra. Abból adódóan, hogy más tartalmat közvetítenek a különböző módon elérhető 2-hosszú utak, megkülönböztettük őket aszerint, hogy a vizsgált csúcs ki- vagy beszomszédjának ki- vagy beélén érjük el őket. Az arányokat itt is meghatároztuk az egyes értékekre vonatkozóan.

4.9.3 További jellemzők

A válaszolási gráf alapján elkészített tulajdonságvektorokat a hozzászólók egyéb jellemző tulajdonságaival egészítettük ki. Ezek között olyan attribútumok szerepeltek, mint például egy fórumozó összes hozzászólásának száma, első és utolsó hozzászólásának ideje, a két időpont között eltelt idő vagy az eddigiek alapján, kétféle módon is kiszámított hozzászólási gyakoriság. Azon hozzászólók, akik mindössze egy alkalommal szóltak hozzá a témához problémássá tették ezt a fajta vizsgálatot, mivel esetükben nem állt módunkban az első és utolsó hozzászólásuk közötti idő kiszámítására. Az ilyen eseteket a tulajdonságvektorban való hiányzóadat-kódok szerepeltetésével oldottuk meg.

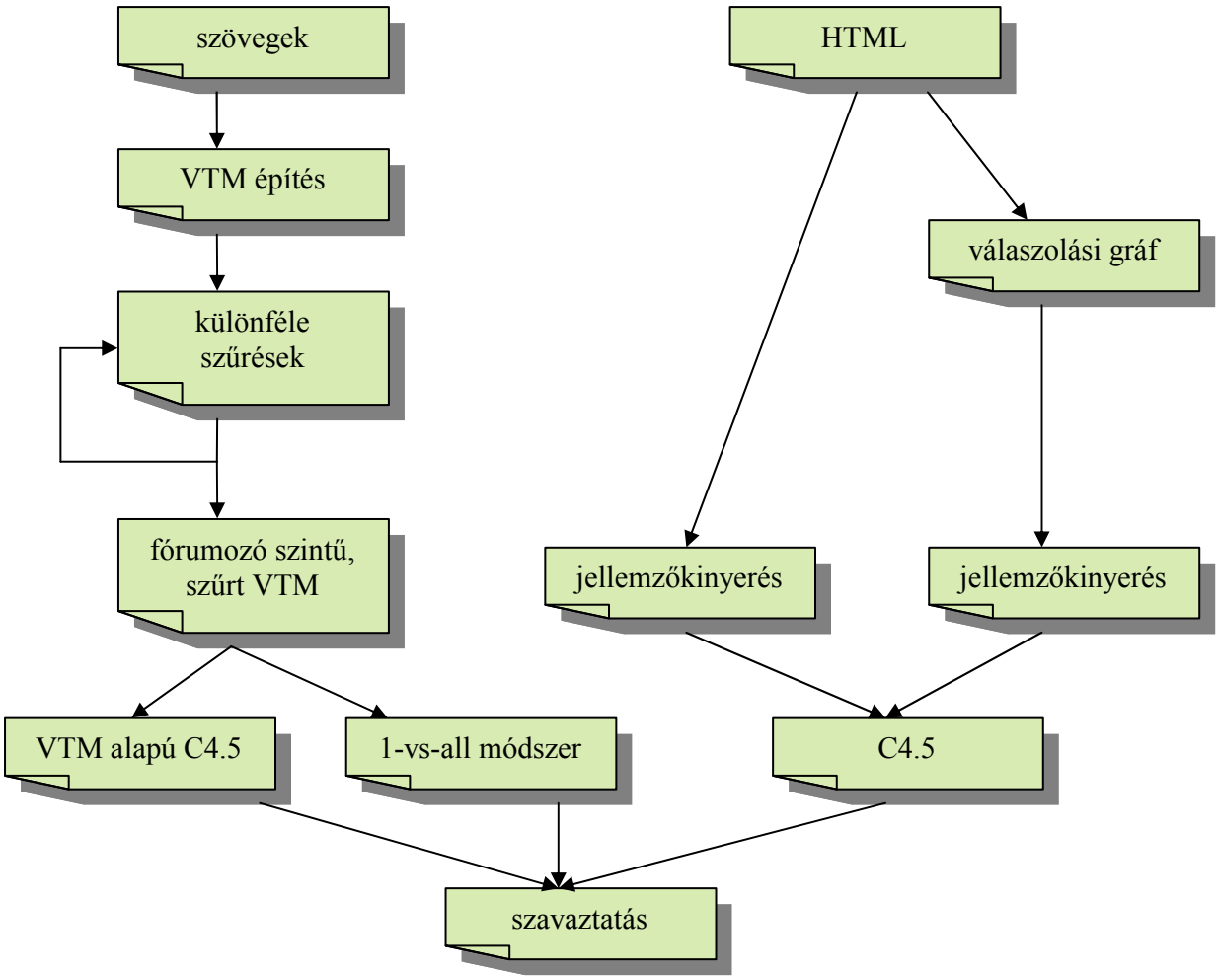
Ezek az ismérvek főképp a SEMMI osztályba tartozás vagy nem tartozás eldöntésének megkönnyítésére használhatók, hiszen azt nagy magabiztossággal állíthatjuk, hogy egy olyan egyénnek, aki sokat és gyakorta szólt hozzá a témához, annak van olyan hozzászólása, amely tartalmaz osztályozható véleményt a népszavazással kapcsolatban.

4.10 A különböző módszerek szavaztatása

A bemutatott három különböző megközelítés (szöveg-alapú 4-osztályos tanulás, szöveg alapú 1-vs-all módszer és a gráfanalízis alapú módszer) teljesen más alapokon nyugszanak, különböző helyeken vétettek, illetve különböző egyedek esetén hoztak helyes döntést. Feltételezhető volt tehát, hogy az eltérő modelleket aggregálva, predikcióik egymással való összehasonlításának segítségével jól használható minták figyelhetők meg a végső osztálycímkék pontosabb meghatározása érdekében. Ennek a feladatnak a megoldására szabályalapú rendszert alkalmaztunk.

Az eredmények ismeretében megállapíthatjuk, hogy a VTM-alapú módszer eredményezte általánosságban a legnagyobb pontosságot. A pontosság azonban tovább javítható, amennyiben amikor az 1-vs-all és a gráfanalízisen alapuló tanulás is azonos osztályba sorol egy egyedet, akkor azt fogadjuk el, függetlenül a VTM módszer predikciójától. A végső osztálycímke meghatározásához még egy hasznos szabályt vettünk föl, ami az ÉRVÉNYTELEN osztály meghatározását célozta. Erre azért volt szükség, mert egyik tanulás sem volt képes ÉRVÉNYTELEN címkére vonatkozó szabályt tanulni, mivel ahhoz túl kevés és túl diverz tanulóadat állt csak rendelkezésünkre. E szerint a szabály szerint akkor kapott egy hozzászóló ÉRVÉNYTELEN besorolást, ha a VTM alapú modell NEM címkével látta el, az 1-vs-all alapján SEMMI típusú volt a hozzászóló, a gráfanalízisen alapuló tanulás pedig valamilyen nem SEMMI típusba tartozást valószínűsített. Ez azzal magyarázható, hogy a NEM osztályt jó arányban felismerő VTM alapú J48-as tanulás NEM címkével látta el, illetve annyiban a gráfanalízisen alapuló döntés is megerősíti ezt a döntést, hogy nem SEMMI osztályba sorolta, a NEM osztályra jellemző szavak alapján döntő 1-vs-all módszer azonban mégsem NEM-nek osztályozta. Jogunk van tehát egy ilyen hozzászólóról feltételezni, hogy egy nem SEMMI, de ugyanakkor nem is NEM típusú egyeddel van dolgunk, a feladatban kitűzött címkék közül pedig az ÉRVÉNYTELEN felel meg ezeknek a kritériumoknak leginkább.

Az egyéni modellek összekapcsolódását, az egész rendszer működését az alábbi ábra szemlélteti:



4.10.1. ábra – A teljes rendszer működése

5. Eredmények

Az eredmények értékelésére kézenfekvő módszer a pontosan osztályozott egyedek és az összes egyed hányadosaként előálló *pontosság* meghatározása. Elképzelhetők azonban olyan alkalmazások, ahol nagyobb jelentőssséggel bír bizonyos osztályok minél pontosabb felismerése, mint a globálisan mért pontosság. Ilyen alkalmazás lehet például egy automatikus minőségellenőrző rendszer, ahol a gyártási folyamat volumenéhez képest jelentkező kisszámú hibás termék azonosítása kritikus jelentőssséggel bír. Az arányaiban kevés hibás gyártmány miatt viszont, ha egyszerűen minden terméket helyesnek osztályozunk kimagasló pontosságot érhetünk el, anélkül, hogy akár egyetlen hibás terméket is helyesen osztályoztunk volna. A népszavazás kapcsán is fontosabbnak tartottuk a SEMMI osztályon kívüli egyedek pontos felismerését.

Ilyen esetekben, ahol az egy-egy osztályon, vagy az osztályok egy részhalmazán mért teljesítményt akarjuk kiértékelni, jobb képet alkothatunk egy osztályozás „jóságáról" az egyes osztályokra vonatkozó *precízó*, *fedés,* valamint *F-mértékek* ismeretében, ezért az eredmények ismertetésénél esetenként ezeket az értékeket is feltüntetjük.

A *c* osztályra vonatkozó precíziót és fedést rendre a

$$precízió_c = \frac{|\{E_c \cap E'_c\}|}{|E'_c|}$$

5.1. képlet – c osztályra számított precízió képlete

, valamint a

$$fedés_c = \frac{|\{E_c \cap E'_c\}|}{|E_c|}$$

5.2. képlet – c osztályra számított fedés képlete

képletekkel számítjuk, ahol E_c a ténylegesen is c osztályba tartozó egyedek halmaza, E'_c pedig a rendszer által c osztályúnak jelölt egyedeké. A *c* osztályra vonatkozó F-mértéket pedig a precízió és fedés harmonikus közepeként értelmezzük[5], azaz:

[5] Létezik az F-mértéknek egy súlyozott változata is, amelynek képletében a precíziót és a fedést különböző súllyal vesszük figyelembe az alkalmazás céljainak megfelelően, mi azonban itt ennek használatától eltekintettünk, azaz a precíziót és a fedést egyforma súllyal vettük figyelembe (a súlyozást megvalósító konstans így az F-mérték képletében sem szerepel).

$$F - mérték_c = 2 * precízió_c * fedés_c /(precízió_c + fedés_c)$$

5.3. képlet – c osztályra számított F-érték képlete

módon számítjuk ki leggyakrabban. Így az F-mértékről elmondható, hogy előnyben részesíti a kiegyensúlyozott modelleket, amelyek mind precízióban, mind fedésben értékelhető eredményt adnak (ellentétben a fent említett szélsőséges esetektől, amelyek magas pontosságot, de minimális – vagy zéró – fedést mutattak).

A kiértékelés során az ún. *leave-one-out* módszert használtuk, vagyis a tanulást úgy végeztük el, hogy az éppen vizsgált egyed kivételével az összes többi címkéjét ismertnek tekintettük. Mivel ez a fajta kiértékelés annyi tanítási fázist eredményez, ahány példa van az adatbázisban, ezt csak olyan esetekben tehetjük meg, amelyekben az alkalmazott tanulóalgoritmus időigénye, illetve az adatbázis mérete ezt lehetővé teszi. Ebben a feladatban relatíve kis adatbázissal, illetve kis időigényű döntésifa-tanuló algoritmusokkal dolgoztunk, ami gyors *leave-one-out* kiértékelést tett lehetővé. Választásunkat itt az a tény is motiválta, hogy az ilyen kiértékelés közelíti a legjobban a teljes címkézett adatbázis felhasználásával tanított modell 1-1 új példán való kiértékelését (összehasonlítva a különböző keresztvalidációs kiértékelési technikákkal).

Esetünkben *baseline* módszer alatt annak az osztályozásnak az eredményességét értjük, amely minden egyedhez a leggyakoribb előfordulással rendelkező osztályt rendeli hozzá. Az alapvető követelmény természetesen ennek az értéknek a túlteljesítése [15]. Esetünkben a leggyakoribb osztály a NEM címkéjű volt, a maga 29 előfordulásával. Az ebből számított baseline pontosság pedig 34,1176%.

5.1 A döntési fa paraméterei

A döntési fával történő tanulás során lehetőség van különböző paraméterek hangolására. A J48-as algoritmus több paramétere is állítható, amelyek közül az egyik a fa leveleire eső egyedek számát határozza meg. A paraméter a WEKA programcsomagban alapértelmezetten a 2 értéket veszi fel, mi ezt 5-re módosítottuk. A nagyobb elemszám az egyes leveleken átlagosan kisebb döntési fákat eredményez, ez segít elkerülni a túltanulást, ami esetünkben a nem túl nagy adatbázisméret miatt fontos szempont volt.

5.2 A különböző egyéni módszerek eredményei

A VTM-modellek építésekor a kezdeti kísérletek során a legjobb 500 tf-idf értékű termre szorítkozva végeztem méréseket, de azok az adatbázis kis méretéből adódóan nem hoztak jó eredményeket. Ezek után a termek gyakoriságának szűrését arra szorítottam meg, hogy csak azok a termek ne legyenek a modell részesei, amelyeket vagy a hozzászólók több, mint 90%-a használt, vagy pedig kevesebb, mint 3 fórumozó hozzászólásában voltak megtalálhatók. A következő táblázat az egyes VTM-alapú modellek eredményeit tartalmazza, a hozzászólások szövegein végrehajtott szűrések, illetve, a termek kialakításánál használt szavak száma (unigram vagy bigram) szerint.

	Unigramok	**Bigramok**	**Uni+bigramok**
Szűrések nélkül	50,5882%	50,5882%	50,5882%
NE-szűrés	51,7647%	48,2353%	52,9412%
Stopword szűrés	56,4706%	45,8824 %	55,2941%
POS szűrés	55,2941%	50,5882%	52,9412%
NE+stopword szűrés	55,2941%	43,5294%	54,1176%
NE+POS szűrés	55,2941%	50,5882%	54,1176%
Stopword+POS szűrés	58,8235%	50,5882%	54,1176%
NE+POS+stopword szűrés	56,4706%	54,1176%	54,1176%

5.2.1. táblázat – A különböző szűrések melletti eredmények

Mivel a stopword- és POS szűrésekkel értük el a legjobb eredményeket (unigramokon használatával), illetve az egyes (unigram, bigram, uni+bigram) eredmények szórása is ezekre az egyik legkisebb, ezért ezt a modellt találtuk a legalkalmasabbnak a további domain-specifikus, *igen/nem/támogat* szavakat tartalmazó mondatszintű szűrés elvégzésére.

	Unigramok	**Bigramok**	**Uni+bigramok**
Domainspecifikus szűrés nélkül	58,8235%	50,5882%	54,1176%
Domainspecifikus szűréssel	51,7647%	55,2941%	58,8235%

5.2.2. táblázat – domainspecifikus szűrés hatása a találati arányra

A további feldolgozások során a két 58,82%-os eredményt elérő VTM közül az uni- és bigramokat egyaránt fölhasználó verziót választottuk. Döntésem azért esett erre a VTM

reprezentációra mert, úgy hittem a bigramok hasznosságában (pl. tagadott szóalak, módosítószó+szóalak párok). Ezt a kiválasztott VTM-et használtam az 1-vs-all tanulásnál is, melynek pontossága pontosságát a következő táblázatban olvashatjuk:

	VTM alapú		
	Precízió	**Fedés**	**F-mérték**
SEMMI	68,4211%	96,2962%	62,50%
ÉRVÉNYTELEN	0,00%	0,00%	NaN
IGEN	47,8261%	50,00%	48,8889%
NEM	54,1667%	44,8276%	49,0566%
Összesített pontosság	58,8235%		

5.2.3 táblázat – a véglegesen fölhasznált VTM modell pontossága

A VTM-en alapuló J48-as tanulás mellett az 1-vs-all és a válaszolási gráfon való szintén J48-as tanulások képezték még a szavazásban résztvevő alrendszereket. Ezek összehasonlítását a következő táblázat tartalmazza:

	1-vs-all módszer			Gráfanalízis		
	Precízió	**Fedés**	**F-mérték**	**Precízió**	**Fedés**	**F-mérték**
SEMMI	47,17%	92,59%	62,50%	50,00%	74,07%	59,70%
ÉRVÉNYTELEN	0,00%	0,00%	NaN	0,00%	0,00%	NaN
IGEN	62,50%	22,72%	33,33%	50,00%	40,91%	45,00%
NEM	58,62%	58,62%	58,62%	66,67%	66,67%	66,67%
Összesített pontosság	55,29%			52,94%		

5.2.4. táblázat – összehasonlító táblázat az 1-vs-all és a gráfanalízis módszere között

5.3 Végső eredmény

A különböző rendszerek szavazásával kapott modell, illetve a tényleges osztályeloszlások összehasonlítása az alábbi tévesztési mátrixot eredményezte:

		ETALON				
JELÖLÉS		**SEMMI**	**ÉRVÉNYTELEN**	**IGEN**	**NEM**	**Összesen**
	SEMMI	27	2	10	7	46
	ÉRVÉNYTELEN	0	4	1	1	6
	IGEN	0	1	9	4	14
	NEM	0	0	2	17	19
	Összesen	27	7	22	29	85

5.3.1. táblázat – a végső osztályozásból előálló konfúziós mátrix

A végső eredmény az egyes osztályokra vetítve illetve összesítve a következő táblázatból olvasható le. A végső eredmény osztálycímkékre vetítve, illetve összesítve a következő táblázatból olvasható le:

Osztálycímkék	Precízió	Fedés	F-mérték
SEMMI	58,69%	100,00%	73,97%
ÉRVÉNYTELEN	66,67%	57,14%	61,54%
IGEN	64,29%	40,91%	50,00%
NEM	89,47%	58,62%	70,83%
Összesen pontosság	67,06%		

5.3.2. táblázat – a végső rendszer eredménye

6. Diszkusszió, további munkák

6.1 Diszkusszió

A vektortérmodell segítségével végzett tanulások eredményeinek megvizsgálásából kitűnik, hogy a szűrések döntő hányada az előzetes várakozásoknak megfelelően javított rendszerünk hatékonyságán. Egyedüli kivételként a kizárólag bigramokon végzett tulajdonnév és POS-szűrések említhetők meg, ahol visszaesés volt tapasztalható a kétféle szűrő külön-külön, illetve együttesen történő használatakor is. Ennek hátterében az állhat, hogy épp a szűrésnek köszönhetően (hiszen először hajtjuk végre a szűréseket unigramokon, majd készítjük el a bigram-statisztikákat) jellegzetesen egymás után előforduló termsorozatokat nem építettünk bele a vektortérmodellbe.

A domainspecifikus szűrést az uni-, bi- és az uni-és bigramos modelleken egyaránt legjobban teljesítő rendszeren kívántam végrehajtani. Az eredmények alapján a stopword- és POS-szűrések együttes alkalmazását választottam[6]. A domainspecifikus szűrés kizárólag az unigramok esetében nem tudott javítani az elért eredményen, ami részben abból fakadhat, hogy ilyen kis méretű tanuló adatbázis mellett az túl erős szűrésnek bizonyult (unigramok használata mellett). Mivel az 1-vs-all csak a biztos esetekben jelöli a nem SEMMI-típusú hozzászólókat, ebből adódóan ezekre az osztályokra precíziója jobb, fedése pedig gyengébb a VTM-nél tapasztaltakhoz képest.

A gráfanalízis eredményeit szemügyre véve, és akár a VTM-alapú, akár pedig az 1-vs-all módszerrel összehasonlítva beigazolódni látszik azon feltevésünk, hogy a válaszolási gráf segítségével javítható a minket főképp érdeklő érdemi tartalommal bíró hozzászólók beazonosításának pontossága. A SEMMI címkéjű egyedeken elért gyengébb eredmény mellett mind a NEM, mind pedig az IGEN címkéjű entitások esetében átlagosan 10-10%-kal jobb F-mértékeket figyelhetünk meg a gráf alapján végzett tanulás esetében.

A három különböző egyéni model (VTM alapú C4.5 illetve 1-vs-all tanulás illetve gráf alapú módszer) habár pontosságban hasonló eredményekete ért el, a predikcióik igen eltérő képet mutattak. Érdemes megjegyezni, hogy minden esetben a SEMMI és a NEM típusú hozzászólók azonosítása járt nagyobb sikerrel, ÉRVÉNYTELEN osztályba sorolt egyedre viszont egyik rendszernek sem sikerült érdemi modellt alkotnia. A leírtak az osztálycímkék aszimmetrikus eloszlásával és az osztályokon belüli diverzitással magyarázhatók.

[6] Megjegyezzük, hogy ezen szűréseknél nem, vagy kevésbé jelentkezik a fentebb említett probléma, hiszen a töltelékszavak szűrésével szemantikailag értelmes szósorozatok kerülnek a szűrés után egymás mellé.

A végső eredmények beigazolják a rendszerek kombinációjára (szavazás) vonatkozó hipotézisünket. Abból eredően ugyanis, hogy a különböző módszerek máshol vétettek, és más helyeken hoztak megfelelő döntést, komoly javulást értünk el, amelyek között kiemelhetjük a SEMMI osztályon elért 100%-os fedést vagy az ÉRVÉNYTELEN címkéjű egyedek felismerésében elért 61,54%-os F-mértéket.

A tanítóhalmazban szereplő igenek és nemek arányától nagyon kis mértékben tér csak el a végső rendszer predikcióinak igen-nem aránya. A korpuszban ugyanis 0,431373 volt az igenek aránya az érvényes szavazatot leadni tervezők arányához képest. Ugyanez az érték a predkiált halmazon 0,424242-nek adódott. Elmondható tehát, hogy amennyiben a fórumra hozzászólást író részsokaságon belül azonos lett volna az igennel és nemmel szavazók aránya, mint a teljes sokaságra nézve, úgy a rendszer sikerrel jósolt volna meg a népszavazás végkimenetelét is.

A végeredmények ismeretében kijelenthető, hogy magyar nyelvű vélemény-detektáló rendszerünk jó eredményeket ért el, hiszen a 34%-os baseline értéket több, mint 30%-kal sikerült felülmúlnunk, a 72,94%-os annotátorok közötti egyetértési szintet pedig a végső rendszer 67,07%-os pontosságával sikerült megközelíteni.

6.2 További teendők

Ahogy az 5. fejezetben láthattuk, igen bíztató eredményeket sikerült elérni, azonban a módszer számos helyen továbbfejleszthető. Ennek egyik módja lehet a hozzászólások sorrendjét (azaz az időbeliséget) is figyelembe vevő tanulás elvégzése a hozzászólási gráf éleinek címkézésével. Ha például a *„Nem értek veled egyet!”* hozzászólás egy egyértelműen bekategorizálható hozzászólótól származó üzenetre érkezett, akkor nagy magabiztossággal állítható, hogy a 2 szerző címkéje egymás ellentettje. A hozzászólások egymásutániságát szem előtt tartva igen hasznos lehetne az anaforák (visszautalásos ismétlések) feloldása is.

Másik lehetőség lehet a korábban ismertetett válaszolási gráf egy eddigieknél mélyhatóbb elemzése, például az oda-vissza élek tüzetesebb vizsgálata. Az előző példához hasonlóan feltételezhetjük, hogy két hozzászóló eltérő címkével látható el, amennyiben nagy súlyú oda-vissza él van az őket gráfban reprezentáló pontok között. Ez ugyanis arra utalhat, hogy nézeteltérésükből fakadóan sokszor reagáltak egymás hozzászólásaira.

A tulajdonnevek kezelése terén is van mód további javulás elérésére, hiszen nyilvánvaló, hogy a *„Nem értek egyet Gyurcsány Ferenc álláspontjával a népszavazás kapcsán.”* és a *„Nem értek egyet az MSZP álláspontjával a népszavazás kapcsán.”* mondatok

ugyanazt a véleményt fejezik ki, hiszen Gyurcsány Ferenc álláspontja megegyezik pártjáéval. Ezért ezeket a mondatokat azonos módon lenne érdemes kezelni a feldolgozás folyamán. Ezt pedig úgy lehet elérni, ha előzetesen elkészítünk egy-egy listát a különböző nézőpontokat képviselő szervezetekről és azok meghatározó alakjairól, majd ezek szövegbeli előfordulásait rendre lecseréljük hovatartozásuknak megfelelően, vagy esetleg az említésre kerülő entitásokat is a hozzászólókkal megegyező módon kezeljük, azaz azok véleményét, hovatartozását is megpróbáljuk automatikusan meghatározni.

7. Összefoglalás

Az automatikus véleménykinyerés a piackutatásokhoz, hírcsoportok vagy fórumok monitorozásához, fogyasztói visszajelzések megismerésére, vagy keresőmotorok informáltságának növelésére egyre gyakrabban használt eszközzé válik, hiszen ilyenkor nagyon fontos a hozzászólók véleményének gyors felmérése. Ez legtöbbször a rendelkezésre álló szöveges információ mennyisége miatt gépi erőforrás igénybevétele nélkül elképzelhetetlen lenne.

Célkitűzésem a kettős állampolgárság témájában hozzászóló fórumozók véleményének detektálása volt. Az elkészült rendszer felhasználásával a későbbiekben lehetőség nyílik majd automatikus véleménygyűjtés segítségével nyert adatok alapján más, gazdasági és politikai témájú kérdések végkimenetelének előrejelzésére. A rendszer további felhasználása értelemszerűen újabb, címkézett adatbázisok készítését is igényli, azonban – mivel méretét tekintve kicsi, gyorsan elkészíthető adatbázissal dolgoztunk – ez nem jelent elvi akadályt a rendszer éles tesztjének, ami például egy aktuális népszavazás várható eredményének vizsgálata lehetne, még a szavazás tényleges időpontja előtt.

A dolgozatban bemutatott munka jelentősége sokrétű. Újszerűsége egyrészt abban rejlik, hogy a korábbiakban még nem készült véleménydetektáló rendszer magyar nyelvre, így elsőként tapasztaltuk meg egy nehéz feladathoz történő korpuszépítés komplexitását. Munkám eredményeként elkészült az első magyar nyelvű véleménydetekciós kutatásokhoz felhasználható korpusz. A korpuszt, mint tanító és kiértékelő adatbázist használva, gépi tanulási algoritmusokat alkalmaztam, amelyek sok, különböző típusú információ kombinációjával az emberi pontossághoz közeli eredményt adtak. Annak ellenére, hogy a rendelkezésre álló adatbázis mérete viszonylag kicsi, zajjal való terheltsége pedig magas volt, az elért végső pontosság megközelíti az emberi egyetértési szintet, a korpusz baseline értékét jelentősen túlszárnyalja.

Mindenen túl a rendszer erősségét jelzi az a tény is, hogy a fórumozókat jól reprezentálják a predikált besorolások, ugyanis a fórumon megnyilvánuló igen és nem szavazók arányát szinte tökéletesen eltalálta, vagyis a fórumhoz hozzászólók véleményét jól reprezentálja.

Bibiográfia

1. R. Metcalfe - *Metcalfe's Law: A network becomes more valuable as it reaches more users.* Infoworld, Oct. 2, (1995), http://www.infoworld.com/cgi-bin/displayNew.pl?/metcalfe/bm050696.htm
2. Barkóczy László: *Az internetes fórumok diffúziós lehetőségei* (2005), http://www.communicatio.hu/mktt/dokumentumok/konferenciak/2005/poszterek/barkoczylaszlo.htm
3. A. Odlyzko, B. Tilly - *A refutation of Metcalfe's Law and a better estimate for the value of networks and network interconnections.* 2005 ACM SIGCOMM Computer Communication Review Volume 35, Issue 5 ,pp. 97 – 100, ISSN:0146-4833 (2005)
4. Bodon Ferenc - *Adatbányászati algoritmusok.* 2002-2008, BME (2008)
5. W. Frawley, G. Piatetsky-Shapiro, C. Matheus - *Knowledge Discovery in Databases: An Overview.* AI Magazine '92 Fall pp. 213-228. ISSN 0738-4602 (1992)
6. Jiawei Han, Micheline Kamber - *Data mining: concepts and techniques (Second Edition)*, Morgan Kaufmann Publisher (2006)
7. Ian H. Witten, Eibe Frank - *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Sys sorozat June, Morgan Kaufmann. ISBN 0120884070 (2005) http://www.amazon.fr/exec/obidos/ASIN/0120884070/citeulike04-21
8. Margaret H. Dunham - *Data Mining: Introductory and Advanced Topics.* Upper Saddle River, NJ, USA, Prentice Hall PTR. ISBN 0130888923 (2002)
9. nrc informatiONline: *Internet penetráció 2007 I. félév*, (2007) http://www.nrc.hu/hirek?&news_id=427&page=details
10. Soo-Min Kim, Eduard Hovy - *Crystal: Analyzing Predictive Opinions on the Web.* Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 1056-1064, Prague, June (2007)
11. Bing Liu - *Web Data Mining.*Springer, ISBN-13: 978-3-540-37881-5 (2007)
12. A.Gulli, A.Signorini - *Building an open source meta search engine.* Special interest tracks and posters of the 14th international conference on World Wide Web, Chiba, Japan, pp. 1004 – 1005, ISBN:1-59593-051-5 (2005)
13. Nozomi Kobayashi, Kentaro Inui, Yuji Matsumoto - *Extracting Aspect-Evaluation and Aspect-Of Relations in Opinion Mining.* Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 1065-1074, Prague, June (2007)
14. Andrea Esuli, Fabrizio Sebastiani - *SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining.* Proceedings of LREC-06, the 5th Conference on Language Resources and Evaluation, (2006), http://tcc.itc.it/projects/ontotext/Publications/LREC2006-esuli-sebastiani.pdf
15. Nobuhiro Kaji, Masaru Kitsuregawa - *Building Lexicon for Sentiment Analysis from Massive Collection of HTML Documents.* Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 1075-1083, Prague, June (2007)
16. *A magyarorszag.hu kettős állampolgárságáról szóló fórumtémája*: http://www.magyarorszag.hu/kapcsolat/parbeszed/agora/nemzet/tema.html?topicid=722

17. Halacsy P., Tron, V. - *Benefits of Resource-Based Stemming in Hungarian Information Retrieval*, LNCS Volume 4730, pp. 99-106 (2007)
18. Kushal Dave, Steve Lawrence, David M. Pennock - *Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews*, WWW '03: Proceedings of the twelfth international conference on World Wide Web, (2003), http://portal.acm.org/citation.cfm?id=775226
19. Kuba, A., Hócza, A., Csirik, J. - *POS Tagging of Hungarian with Combined Statistical and Rule-based Methods* in Proc. of the Seventh International Conference on Text, Speech and Dialogue (TSD 2004), Brno, Czech Republic pp. 113-121 (2004)
20. *Free Stop Word Lists in 23 Languages:* http://www.semantikoz.com/2008/04/02/free-stop-word-lists-in-23-language (2008)
21. Ethem Alpaydin: *Introduction to Machine Learning..*, ISBN-13: 978-0-262-01211-9 (2004)
22. Quinlan, R.C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA. (1993)
23. Witten I. H. and Frank E. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, second edition. (2005)
24. Batagelj V., Mrvar A.: Pajek - *Analysis and Visualization of Large Networks*. in Jünger, M., Mutzel, P., (Eds.) Graph Drawing Software. Springer, Berlinp. 77-103. (2003)