Tudományos Diákköri Konferencia

Szemantikus nyelvi elemzés alkalmazása döntéstámogató rendszerekben

Soós István (IV. Inf)

Konzulensek:

Prof. Vámos Tibor (MTA SZTAKI)
Dr. Charaf Hassan (BME AUT)

Tartalomjegyzék

Összefoglalás

Dolgozatomban a magyar nyelvű számítástechnikai szövegfeldolgozás problémáit és lehetséges megoldásait tárgyalom egy potenciális felhasználási terület, az elektronikusan támogatott közigazgatás és az önkormányzatok döntéstámogatásának vizsgálatával.
A fő kutatási területem a szemantikus szövegértelmezés, de a terület maradéktalan bemutatásához az alapoktól szeretném kezdeni az elemző rendszer működését.

A dolgozat második fejezetében röviden vázolom az elektronikusan támogatott önkormányzatokkal szemben támasztott igényeket és speciális feladatokat, különös tekintettel a magyar nyelvű környezetre.

A harmadik fejezetben megvizsgálom a számítástechnikai szövegfeldolgozás általános problémáit, melyekről általában kevés szó esik, de a feladat fontos technikai részét képezik.

A negyedik és ötödik fejezetben mutatom be a különböző mélységű szövegelemzés feladatait és az ezekre kifejlesztett megoldásainkat. A negyedik fejezetben a szintaktikai elemzésről, a szabályokra épülő, pontos nyelvi jelentéstartalomtól általában független elemzési módszerről írok. Az ötödik fejezetben bemutatom az általunk kifejlesztett különböző szintű szemantikai elemzés módszereit és felhasználásait.

A fejezetek tárgyalása során a számítástechnikai feladatokat általában a megoldás módszerének rövid vázlatával együtt tárgyalom, általában a technikai megvalósítás részletei nélkül. A hatodik fejezetben bemutatom az elkészített szoftvereket, vázolom a felépítésüket és működésüket.

Végül a hetedik fejezetben bemutatom azokat az elképzeléseket és irányvonalakat, amelyeken elindulva a rendszer továbbfejlesztését tervezzük.

1. Bevezetés

A szintaktikus (szabályalapú) és szemantikus (jelentésalapú) nyelvi megértés a számítástechnikai nyelvészet egyik legrégebbi és legösszetettebb problémája. Természetes nyelven leírt szövegeknél nehéz meghatározni, hogy egy szöveg miről szól, mi a tartalmi mondanivalója. A magyar nyelvhez és nyelvtanhoz hasonló bonyolultságú nyelvekre és nyelvtanokra ez különösen igaz: máig nem írt senki olyan programot, amely egy tetszőleges, nyelvtanilag helyes mondatot valamilyen módon teljesen elemez.

Bár a probléma nem megoldott, egyre több eljárást dolgoztak ki a megoldás megközelítésére. A módszerek egy része a szintaktikus elemzésre koncentrál: hasonló témájú, gyakran megegyező szófordulatokból álló szövegek elemzésével és feldolgozásával foglalkoznak. Ilyen például a szegedi Kalmár-laboratórium kutatása gazdasági szövegek, tőzsdei gyorsjelentések információtartalmának kiemelésével.

Az MTA Számítástechnikai Automatizálási Kutatóintézetén [1] belül, a kaposvári elektronikus önkormányzati projekt keretében kidolgozott elemző módszerünk kevésbé hagyatkozik a szintaktikus elemzésre, sokkal inkább a szemantikus jelentéstartalomra koncentrál. A módszert és részleteit egy az önkormányzati ügyintézőknek szánt döntéstámogató rendszerhez alakítottuk ki, de általános leírásaival természetesen máshol is alkalmazható.

Az utóbbi időben különböző pályázatok és fejlesztési feladatok kapcsán egyre divatosabb kifejezés „a közigazgatási rendszer elektronizálása” vagy az „elektronikusan támogatott önkormányzati ügyintézés kialakítása”. A feladat társadalmilag, szociológiailag és számítástechnikai szempontból is többszörösen összetett. Dolgozatomban megpróbálok egy átfogó képet adni a problémák néhány számítástechnikai – és hozzá szorosan kapcsolódó szociológiai – részéről, s bemutatom, hogy milyen módon integráltuk a korábban említett kutatási részterület, a szemantikus nyelvi megértés eredményeit egy ilyen döntéstámogató modellbe.

Szeretném kiemelni két partnerünket, s egyúttal köszönetet mondani segítségükért: a MorphoLogic [2] és a Kalmár László Kibernetikai Laboratórium munkatársaival folytatott beszélgetéseink, az általuk adott ötletek és problémafelvetések mindig ösztönzőleg hatottak ránk. A később részletezett MorphoLogic által írt elemző program nélkül munkánk reménytelenül nehéz lett volna.

2. Közigazgatás elektronikus támogatása

Az Egyesült Államok példáját követve többek között az Európai Unió is fontos hosszú távú stratégiai fejlesztésnek tartja a közigazgatás és a helyi önkormányzatok elektronikus támogatását és tehermentesítését. Az EU normáit követve Magyarország is elkezdte saját rendszerének kiépítését, aminek egyik első kísérleti terepe a Nemzeti Kutatási és Fejlesztési Pályázatok által támogatott kaposvári elektronikus kormányzási modell lett.

2.1. Mire kell figyelni?

Egy ország (vagy egy kisebb önkormányzat) napi ügyeit teljesen átszövő rendszer esetében körültekintően kell eljárni mind szociális-társadalmi, mind technológiai felületének és felépítésének megtervezésekor. Egy tényező figyelmen kívül hagyása vagy rosszul megválasztott értelmezése a teljes rendszer megítélését elronthatja.

Korosztály-problémák

Az elektronikus közigazgatási rendszerek egyik legfontosabb célkitűzése, hogy ne csak a fiatalabb, informatikai területen nagyobb jártassággal rendelkező korosztály tudja használni. Fontos, hogy az idősebb emberek is találjanak módot a kezelésre. Egy 2004. tavaszi felmérés alapján az idős emberek jelentős részének van a családon belül olyan közeli rokona, aki informatikailag jobban képzett, s tud számára segítséget nyújtani.

Elérhetőség biztosítása

Egy másik fontos szempont, hogy a társadalom minél szélesebb rétege elérje azt a közeget (Internet), amin keresztül ezek az úgynevezett E-government rendszerek kommunikálnak. Hazánk jelentős lemaradásban van a fejlett országokhoz képest, így jelentős kormányzati energiát fordítanak az Internet minél szélesebb körű elterjedésének segítésére.

Bizalom

Elektronikus szolgáltató rendszereket általában kevés bizalommal közelítünk meg, s csak a hosszú használat során kialakult pozitív tapasztalat javíthat ezen. Jó példa erre a hazai bankok home banking rendszerei: évek óta működnek, kritikus hiba nem került elő a bevezetésük óta. Mégis: csak az utóbbi évben kezdték el tömegesen használni, előtte csak a számlainformáció-lekérdezésig terjedt velük szemben a bizalom.
Egyetlen rossz tapasztalat bármikor többet ronthat a rendszer megítélésén, mint sok év stabil működés.

Használhatóság

Elektronikus közigazgatási rendszerek esetében nagyon fontos, hogy a felhasználó minden olyan segítséget megkapjon, amit a személyes ügyintézés során megszokott. Nem célszerű beadványok és kérdőívek tömegével szembesíteni, s azt sem várhatjuk el

mindenkitől, hogy pontosan ismerje panaszának illetve kérelmének pontos törvényi meghatározását.

A személyes meghallgatásokhoz hasonlóan célszerű egy olyan felületet is biztosítani, ahol a felhasználó a saját szavaival megfogalmazott beadványát el tudja juttatni az illetékes ügyintézőknek. A rendszer inkább kérdezzen vissza, hogy az ügyfélnek pontosítania kelljen a beadványt, mint rosszul értelmezve azt – további bosszúságot okozva – az elektronikus rendszer megítélést rontsa.

2.2. Magyarország speciális helyzete

Több kész elektronikus megoldás is található a számítástechnikai piacon, mégsem tudjuk a „teljes termék a dobozban” elvet követve átvenni és alkalmazni őket. Ennek elsősorban törvényi és joggyakorlati, másodsorban nyelvi akadályai vannak.

Törvények és a jogrendszer

Kész informatikai megoldások eddig elsősorban angolszász nyelvterületre, angolszász jellegű törvényhozási múlttal rendelkező országokban készültek. Jogrendszerük precedens alapú, ellentétben a Magyarországon is használt porosz jellegű törvényhozási és bíráskodási gyakorlattal. A precedens alapú jogrendszer informatikai előnye – a rutinesetek gyors és hatékony feldolgozása – Magyarországon (és az Európai Unió legtöbb országában) nem használható.

Probléma még a törvényhozás elektronikus tájékoztatási kultúrájának hiánya is: ugyan nagyon sok törvény és jogszabály megtalálható az Interneten, de a későbbi módosítások csak ritkán, s ha meg is vannak, általában az eredeti szövegkörnyezettől nagyon elkülönült formában.

A rendszer tervezésekor megoldandó probléma még az elektronikus aláírás szűk körű elérhetősége, illetve az adatvédelmi törvénynek megfelelő informatikai rendszer kiépítése.

Nyelvi problémák

A magyar nyelv a számítástechnikailag legnehezebben feldolgozható nyelvek egyike. Ragozása agglutináló, mondatszerkezete szabad, szavak felcserélésével a mondat hangsúlya és mondanivalója teljesen más fordulatot vehet. Teljes elemzőt még senki sem írt rá, a terület alapkutatása több helyen is nagy erőforrásokkal folyik. Automatizált vagy automatizálható nyelvi eszközök jelenleg csak nagyon korlátozott feladatokra állnak rendelkezésre.

3. Szövegfeldolgozás

Kutatásunk során csak a számítástechnikailag megoldható problémákra koncentráltunk, feltételeztük, hogy nekünk már csak elektronikus dokumentumokkal kell foglalkozni, nem törődünk azok keletkezési körülményeivel. Ez lényeges egyszerűsítés a bevezetőben vázolt feladatokhoz képest, de lehetővé teszi a problémára koncentrált feladatmegoldást.

Elektronikus szövegek sokféle formátumban létezhetnek. Eltérő operációs rendszerek eltérő nyelvi beállítások és programok más és más formátumban kezelik és tárolják a dokumentumokat. Bár az elemzés szempontjából nem kutatási feladat, de ezeket a szövegeket a szöveg vizsgálata előtt szabványos formátumra kell hozni. Ennek technikai részleteiről szól röviden ez a fejezet.

3.1. Ékezetek, kódkészletek

3.1.1. Unicode

A számítógép képernyőjén, a megnyitott fájlokban megjelenő szövegek nem a hagyományos értelemben vett karaktersorozatok. Számítástechnikai értelemben csak számok sorozata, melyeket egy szabályrendszer, egy úgynevezett karakterkód-kiosztási táblázat alapján olvasható karakterekké alakítunk.

A számítástechnika kezdeti elterjedése után hagyományok alapján az ASCII kódtábla volt az egyeduralkodó. A különböző nyelvterületekre írt nem angol nyelvű alkalmazások azonban megkövetelték az adott nyelv speciális betű- és írásjeleit, így minden főbb nyelvterületre kialakult egy saját kódtábla, mely az adott nyelv kívánságait teljesíti.

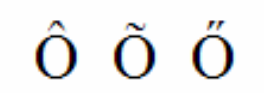
A kezdeti kódtáblák még 7 illetve 8 bites kódkészlettel dolgoztak, ami azt jelentette, hogy minden betű belefért egyetlen bájtnyi információba. Keleti (arab, kínai, koreai, japán) nyelvterületeken azonban ez a 256 érték nem elegendő, így kialakultak speciális, általában 16 bites kódtáblák is. A kódtáblák káoszát mindenki el szeretné kerülni, ezért kialakult egy egységes, szabványosnak számító kódtábla, a Unicode [3]. A Unicode több szabványt is magába foglal, a legelterjedtebb közülük az UTF-8, amely változó bitszámú kódkészlettel tetszőleges karakter ábrázolására alkalmas.

A mai fejlett számítástechnikai platformok, fejlesztői környezetek legtöbbje támogatja, sőt erőteljesen javasolja a Unicode használatát, így mi is ebben a formátumban dolgozunk a szövegekkel. Ami esetleg nem Unicode formátumban érkezik, azt erre az alakra kell hozni.

3.1.2. Magyar ékezetek

Magyar ábécé támogatására elsősorban az ISO-8859-2 és a Windows 1250-es kódtábla-kiosztása használatos. A kódtáblák Unicode formátumra alakítása nem bonyolult, leszámítva a hosszú, dupla ékezetes betűinket, azaz a következőket: ő, ű, Ő, Ű. Mindegyiknek van kalapos, hullámvonalas és dupla hosszú ékezetes változata. Mivel a Unicode szabvány alapján az utóbbit kell használni, ezért bármilyen forrásból érkezett

a szöveg, először meg kell vizsgálni, hogy a felsorolt „hibás” ékezetekből tartalmaz-e valamit, s ha igen, akkor azt minden további feldolgozást megelőzve megfelelő formátumra kell hozni.



1. ábra: az „Ő” betű változatai

3.1.3. Ékezetmentes szövegek ékezetesítése

Elektronikus dokumentum lehet ékezetes vagy ékezetmentes e-mail, Word-dokumentum, normál szövegfájl, s még rengeteg – fel nem sorolható – formátum. Fontos azonban kiemelni, hogy egy nyelvi elemző hatékony működéséhez ékezetes környezet szükséges. Ha a dokumentum nem tartalmaz ékezeteket, akkor az elemző program feladata, hogy a szöveget helyesen értelmezve ékezetekkel egészítse ki azt.

Ha a szöveg beolvasása során találkozunk ékezetes karakterekkel, akkor megnyugodhatunk: a dokumentum készítője valószínűleg az egész szöveget számunkra megfelelő módon, ékezetesen írta. Ha azonban egyetlen ékezetes karakter sincs benne, akkor felmerül a gyanú, hogy valamilyen okból nem használta a teljes magyar karakterkód-kiosztást. Bár előfordulhat olyan eset is, amikor a szövegben egyetlen ékezetes karakter sem található, ékezetes karakter nélküli szövegeknél – a legrosszabb lehetőséget feltételezve – mindig előírjuk a szavak lehetséges módokon történő ékezetesítését.

Az ékezetesítés során először az összes lehetséges ékezetes változatot képezzük (pl. „ol” esetében ól, öl, ől), majd a szótanilag értelmezhetetleneket („ol”, „ől”) elhagyjuk.

3.2. Szöveg darabolása

Szövegfeldolgozáskor sajnos nem számíthatunk arra, hogy a feldolgozandó szöveg szépen szerkesztett, jól tagolt, elírásoktól mentes, s egységes formátumot követ. Ezt mindenképpen egységesíteni kell, a hibás egybeírásokat külön kell választani, az írásjeleket, a segédkaraktereket el kell különíteni a szövegtől.

Elemző rendszerünkben az írásjelek és szünetek (szóköz, sortörés) mentén daraboltuk a szöveget, s a közöttük elhelyezkedő részeket a szöveg szavainak tekintettük. Figyeltünk arra, hogy Internetes kifejezések (pl.: URL: http://www.sztaki.hu/) esetén ne daraboljunk a pontok mentén, de rosszul írt mondatkezdések esetén (pl.: „elment.Ekkor”) szétválasszuk a szavakat.

A darabolás eredménye egy szavakból és írásjelekből álló lista, melynek speciális szerkezete az egyes pozíciókat szóhalmazként kezeli. A darabolás során eltároljuk az eredeti szövegben elfoglalt pozíciókat is, így egy interaktív rendszer pontosan tud visszautalni rá.

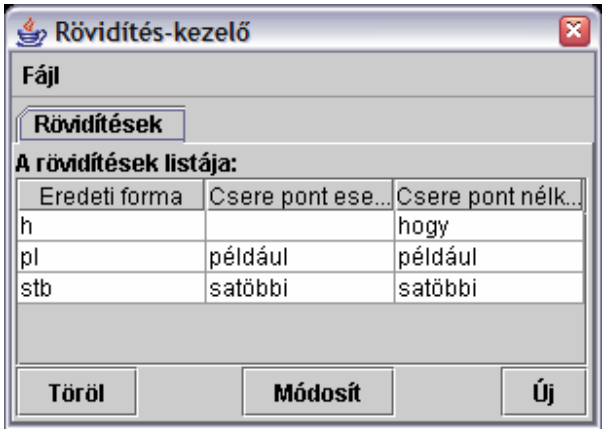
3.3. Rövidítések feloldása

A szöveg darabolása során kaptunk egy szó- és írásjellistát. A szerkezet nem alkalmas rövidítésekhez tartozó, s mondatvégi pontok megkülönböztetésére, így célszerű ezeket az elején elkülöníteni.

A megvalósított programban kétfajta rövidítést kezeltünk:

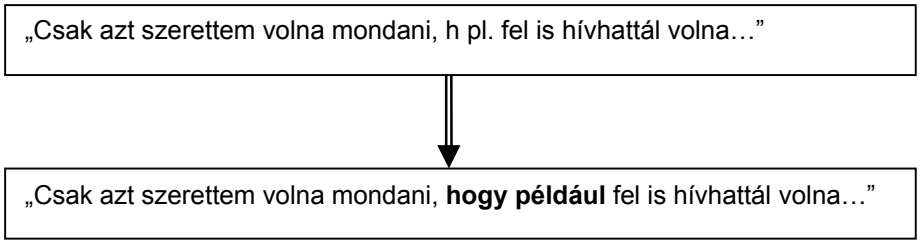
- ponttal jelzett rövidítés
- pont nélküli rövidítés

Utóbbi esetben a rövidítés feloldásához nem szükséges a rövidített kifejezés utáni pont, így pongyolán megfogalmazott szövegek esetén is helyesen kezelhetjük őket.



2. ábra: Rövidítések feloldása: adminisztrációs felület

Egy egyszerű példa a feloldás működésére, a szöveg értelmét most figyelmen kívül hagyva:



3. ábra: Rövidítések feloldása működés közben

4. Szintaktikai elemzések

A szövegfeldolgozás technológiai jellegű bevezetése után rátérhetünk a konkrétabb elemzésekre, melyek kutatási feladataink részét képezték.

Szintaktikai elemzésnek azokat az elemzéseket nevezzük, amelyek valamilyen szabályrendszer alapján vizsgálják a szöveget, módosítják, kibővítik azt, esetleg új struktúrába szervezik. Az elemzési lépések egyszerűek, a szabályrendszer általában nagy, de automatizált vagy fél-automatikus bővítése könnyen támogatható számítógépes programmal.

4.1. Szavak elgépelésének ellenőrzése

Az egyik leggyakoribb elgépelési hiba a magyar és angol billentyűzetkiosztás váltása közben az „y” és a „z” felcserélése, de ezen kívül több száz más kifejezést is gyakran elgépelünk. A javítás egy módja, ha a tipikus elgépelési hibákat egy táblázatba rendezzük, s a hibás szavakat a helyes változatukra cseréljük.

Az elkészített programmodulunk ennél kicsit tovább megy: minden egyes elgépelési változathoz akár több helyes javítási változatot is megadhatunk (ékezettel kapcsolatos elgépelési hibáknak több ékezetes javítása is lehet). Ha a szövegben az adott – feltételezhetően hibás – szót találja, akkor lecseréli azt az összes javított változatára. (A szavakat az egyes pozíciókban szóhalmazként kezeljük, az adatszerkezet működését a 6.2.3-as fejezetben részletezem.)

4.2. Szavak morfológiai elemzése

A magyar nyelv agglutináló szerkezetű, azaz a toldalékokat a szótő után téve, esetleg a szó tövét is módosítva képezzük. Egy szót csak karakterek sorozataként tudunk értelmezni egészen addig, míg meg nem mondjuk a szófaját, szótövét, ragozási tulajdonságait. A morfológiai modul feladata, hogy mindezeket minden egyes szóra meghatározza.

Egy ilyen modul megírása önmagában is szép feladat lenne, de szerencsére a MorphoLogic már készített egyet, s a rendelkezésünkre bocsátotta programját.

4.2.1. HuMor

A morfológiai elemzéshez a MorphoLogic HuMor elnevezésű programját használjuk. Magyarországon ők voltak a nyelvészeti programok egyik legelső fejlesztői. Az általunk használt régebbi változat csak parancssoron keresztül kommunikál, ezért szükséges volt egy illesztő felületet készíteni hozzá.

Egy példa a program működésére:

```
Morphological analyser, stemmer and speller (C) MorphoLogic, 1998
Version 1.1
->bolygó
bolygó[MN]+[NOM]
bolygó[FN]+[NOM]
bolyog[IGE]=bolyg+ó[MIF]+[NOM]

->kedves
kedves[MN]+[NOM]
kedv[FN]+es[SKEP]+[NOM]

->fiaiéi
fia[FN]+i[IKEP]+éi[POSi]
fi[FN]+ai[PSe3i]+éi[POSi]
fiú[FN]=fi+ai[PSe3i]+éi[POSi]
```

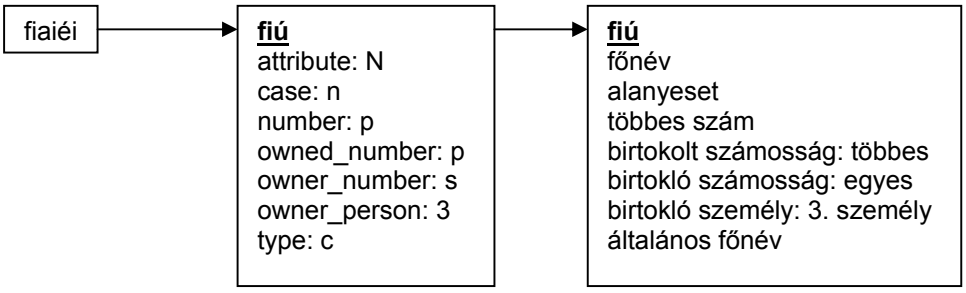
4. ábra: A MorphoLogic programja futás közben

Minden egyes szóhoz egy vagy több elemzési lehetőséget is megad, a szórészletek szerepét egyértelműen jelölve. Ezeket az elemzéseket felhasználva a szavak morfológiai elemzését egy szabványosnak tekinthető leíró formátummal kezeljük a továbbiakban. A formátum rövid megnevezése a következő fejezetben bemutatott MSD:

4.2.2. MSD kódrendszer

Az MSD (MorhoSyntactic Description) [4] egy egységes, több nyelvre kiterjesztett morfológiai leíró nyelv. Egy Európai Unióhoz kapcsolódó projekt, a MULTEX-East részeként definiálták, angol és kelet-európai nyelvek szavainak jellemzéséhez használható. Szavakhoz rendel különböző attribútumokat, melyek szótő megadása esetén teljesen leírják a szavak ragozott alakját.

Az előző példában említett „fiaiéi” szó MSD leíróval jelölt elemzése:



5. ábra: MSD leírás és értelmezés

Minden attribútum egy részinformációt rejt az egész szóval kapcsolatban, de a szótővel együtt egyértelműen leírják a szó eredeti formáját.

Az MSD kódrendszert a Kalmár-laboratórium mintájára vezettük be. Ők létrehoztak egy körülbelül egymillió szavas nyelvi korpuszt, melyben különböző helyről (újság, szépirodalom, gyermekszövegek stb.) vett szövegeket elemeztek. Minden egyes szóra

ellenőrzött módon meghatározták a nyelvtani elemzést, s a teljes adatbázist rendelkezésünkre bocsátották. Az adatbázis egy részét integráltuk rendszerünkbe.

HuMor -> MSD átalakítás

Természetesen a kétféle kódolás között meg kell oldani a konverziót, amit egy egyszerű szabályrendszerrel végzünk el: a szavak szórészleteinek elemzési címkéiből levágjuk a kiegészítő attribútumokat egészen addig, amíg érvényes szótőelemzést nem kapunk. Ha a címkék levágása után a maradék rész címkéi nem alkotnak értelmes szótövet (vagy már a HuMor sem ismert fel benne semmilyen részletet), akkor ismeretlennek jelöljük meg a szót.

Az ábrán a szabályrendszer adminisztrációs felülete látható:

6. ábra: HuMor-MSD konverziós szabályok

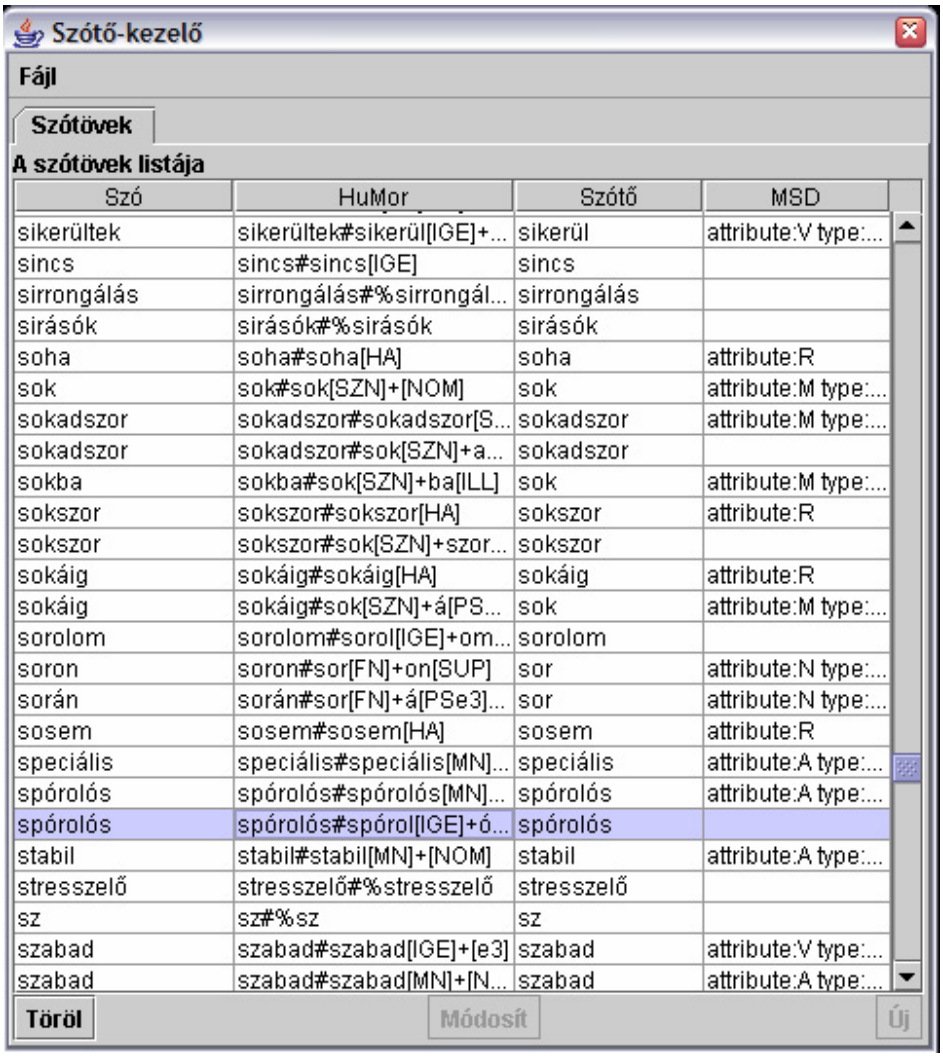
4.2.3. Gyorsítótár

Az elemzés gyorsítására célszerű, hogy a szövegek gyakran előforduló szavainál ne kelljen újra és újra az elemző modulhoz fordulni, hanem a memóriában eltárolt eredményekkel dolgozzunk. Erre a célra létrehoztunk egy gyorsítótárat (cache), s elemzéskor először ehhez fordulunk. Ha a kért szó nincs benne, akkor futtatjuk csak le a HuMor elemzőt, s a kódkonverziós szabályokat.

Bizonyos valósidejű alkalmazások megkövetelik, hogy a memóriát korlátosan kezeljük, a gyorsítótár ne növekedjen a végtelenségig. Létrehoztunk egy memória-korlátos változatot is a következő működéssel:

- Minden elemzési értéknek van egy öregségi értéke
- Minden hozzáférés során ezt az értéket felezzük
- Minden elemzési ciklus után ezt növeljük
- Minden elemzés után az öregségi értékek szerint növekvő sorba rendezve csak az adott darabszámú „legfiatalabbakat” tartjuk meg.

A gyorsítótár tartalmát a futás során ellenőrizhetjük, később elmenthetjük, visszatölthetjük. A következő ábrán egy ilyen részletet láthatunk.



7. ábra: Szavak morfológiai elemzése

4.3. Szinonimák a szövegben

Beszéd közben nagyon sok olyan szót és szófordulatokat használunk, melyek jelentéstartalma nagyon hasonló. Ezek a szinonimák vagy szinonim jellegű szavak jelentősen nehezítik a szövegfeldolgozás számítógépes működését, hiszen mi ugyan beszéd közben ugyanarra a fogalomra asszociálunk, az elemző megkülönbözteti a külalakjukat, ezáltal a jelentéstartalmat.

A szinonima-kezelő modul feladata, hogy az elemzés során ezeket a hasonló jelentésű szavakat összemossuk, s az elemző további részei számára átlátszóvá tegyük őket. Ennek megoldására egy szinonimaszótárt alkalmazunk, mely megadja:

- A szinonim szavak csoportjait

- A csoportok tagjainak (egységes) szófaji jellemzőit, azaz milyen szófajú, milyen ragozású esetekben tekinthetjük szinonimának őket
- A csoport tagjainak átjárhatósági korrekciós értékét (minden esetben kisebb, mint 1)

A modul a szöveg összes szavára megpróbál szinonim jelentésű szavakat illeszteni, s egyezés (szó, szófaj és ragozás) esetén új szóval bővíti az adott pozíciókat. Az új szó eredetiségének mértékét az eredeti szó és az előbbi korrekciós érték szorzata alapján számolja ki (így az eredetiség minden esetben csökken).

```
Pl. a vár szó esetén:
eredetileg: vár (ige, 1), vár (főnév, 1)
A bővítés után: vár (ige, 1), vár (főnév, 1), kastély (főnév, 0,9)
```

4.4. Szövegek nyelvtani elemzése

4.4.1. Nyelvtani elemzési módszerek

Az utolsó fél évszázadban részben a számítástudomány fejlődésével összekötve a nyelvészet is új utakra indult. A generatív nyelvészet mély struktúrákat igyekezett levezetni, amelyeket a fejlődés tovább gazdagított a nyelvpszihológia és nyelvtörténet eredményeivel. Kiemelte a mondat összefüggéseire utaló topic-ot és a lényeges mondanivalót hordozó fókuszt. (Részletesebben lásd az Új magyar nyelvtan [5] című könyvet.)

Az általunk alkalmazott módszerben ezek az elvek is szerepelnek egyszerűsített formában, közelebb a hagyományos nyelvtanhoz. Mivel a fejlesztés kezdetén a feladat nyelvi környezete elég pontosan körülhatárolt volt, ezért a bonyolultabb elemzési változatokra nem volt szükség. Gyakorlati megközelítésünket az Occam-i borotva elv vezette, miközben a későbbi alkalmazások során nem zárjuk ki a mélyebb szemantikai értelmezéseket sem.

Technológiai értelemben nyelvtani elemzésen a szöveg szavaiból felépíttet hierarchikus szerkezetet értjük. Az elemző feladata, hogy az elemzés kezdetén kapott listát, melyben szóhalmazok (a szavak, ékezetesített, illetve szinonim változataik) követik egymást, erre az előírt formátumra hozza.

4.4.2. Nyelvtani szabályok

Az elemzés során egyszerű, gyorsan iterálható szabályokat szeretnénk használni, ezért a következő nyelvtani szabályrendszert dolgoztuk ki:

- Egy szabály egymást követő szavakon értelmezett művelet. Egy szabályt csak akkor alkalmazunk, ha az általa előírt összes szó jelen van és megfelelően illeszkedik.
  Ha a szabály alkalmazható, akkor minden esetben alkalmazzuk.

- A szabály alkalmazása esetén az elemzés párhuzamossá válik: megőrizzük az eredeti szerkezetet is, de a szabály által lefedett szavakat „összehúzzuk” egyetlen szóba, a szerkezet szabályba bevonható szóvá válik.

- Az összehúzásnak (pontosabban a szabálynak) mindig van egy főeleme, amivel helyettesíthetjük az összehúzott elemeket. A főelemet kiegészíthetjük (a szabálynak megfeleltetett) további MSD attribútumokkal.

- Az egymást követő szavak illeszkedését a szótövek és/vagy az MSD attribútumok megfeleltetésével vizsgáljuk. Így kialakíthatóak nyelvi környezettől független szabályok (pl. a főnév és az előtte álló névelő), illetve az adott szó nyelvtani szerkezetétől függő szabályok, a ragozási esetekkel együtt.

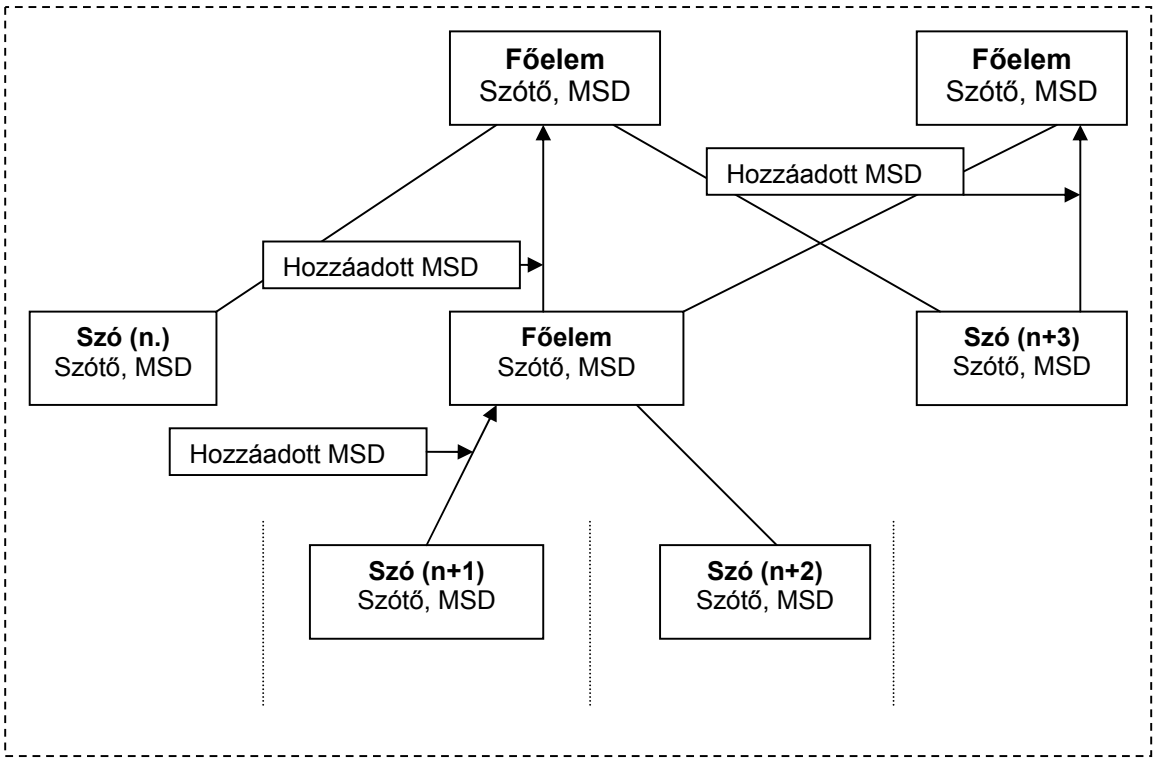
A szabályrendszerrel könnyen leírhatóak egyszerű nyelvtani szerkezetek, s lehetőséget biztosít a bővítésre is. Jelenlegi rendszerünk nem tartalmaz sok szabályt, csak egy felületes nyelvtani elemzésre alkalmas, de már ez is elegendő strukturáltabb mondatkép kialakítására.

Fontos szempont, hogy a szabályrendszerünk nem korlátozódik mondatokra, azaz lehetőség van a mondatokon túlnyúló, mondatokat struktúrába szervező szabályok készítésére is. A lehetőség jelenleg csak elméleti, mert első lépésben a mondatokat kellene egy szabályrendszerrel lefedni, s csak utána következhet a mondatok közötti kapcsolatok vizsgálata.

4.4.3. Elemzési eredmény

A fent leírt szabályokat a szövegre iteratívan alkalmazva eljutunk arra a pontra, amikor már nem keletkezik több szabály. Ekkor a nyelvtani elemzés befejeződik.

Az ábrán a nyelvtani szerkezetek képzésének egy egyszerű sematikus ábrázolása látható:



8. ábra: A nyelvtani elemzés eredménye

Minden egyes szó lehet főelem vagy kiegészítő elem, szabálytól függően. Ha egy szó egy szabályon belül főelem, más szabályon belül betölthet más szerepet is, nem kötelező érvényű a főelem tulajdonság megtartása. Ezáltal egy sokszorosan összetett, párhuzamos elemzési fa alakul ki: egy adott szövegrésznek több (rész-) elemzési képe is létrejön. A főelemek lefedik az alájuk tartozó struktúrát, helyettesítik azt, de tetszőleges részlete más struktúrákban is felhasználható marad.

Az így kapott struktúrákat kétféleképpen rendezve vizsgálhatjuk:

- Az alkalmazott szabályokat kezdőelemenként azonosítjuk (nyelvtani elemzési fa rajzolásához használható, hiszen a mondatok kezdetétől indulva leszámolhatjuk a legnagyobb fedésű ágakat)

- Az alkalmazott szabályokat a főelem alapján rendszerezve azonosítjuk (állítások részstruktúrájának keresésére alkalmas)

5. Szemantikai elemzések

Szemantikai elemzésnek azokat az elemzéseket nevezzük, amelyek valamilyen nyelvi tulajdonság, jelentéstartalom alapján vizsgálják a szöveget, összefüggéseket, állításokat keresnek benne. Jellemzőjük, hogy a tudásbázis nehezen alakítható ki automatikus módon, gyakran komoly szakértői munka kell hozzá.

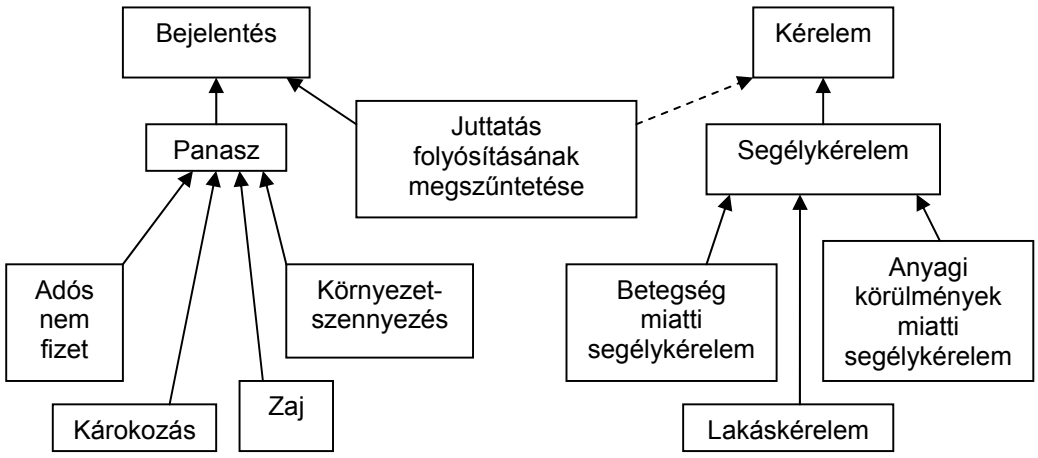
A felsorolt feladatok különállóan is értelmezhetőek, de egymásra épülésük nagyon szoros. Szemantikai elemző rendszereinknél feltételeztük, hogy a szöveg szavait morfológiailag elemeztük, s a kategóriák elemzésének kivételével a nyelvtani elemzés megléte is szükséges.

5.1. Kategóriák

Természetes kommunikáció során (telefonbeszélgetés, újságcikk, könyv stb.) a párbeszéd és szöveg témáját bizonyos kulcsszavak alapján meg tudjuk határozni. Az első mondatoknál még nem biztos, de amint előbbre haladunk a szövegben, egyre inkább lesz egy áttekintő képünk a témáról, el tudjuk azt helyezni saját fogalomrendszerünkben. Terminológiánkban ezt a tematikus besorolást neveztük el kategória-elemzésnek, kategória-besorolásnak.

5.1.1. Kategória-fa

A kategóriákat egy fa jellegű struktúrába szervezzük. Annyiban különbözik egy fától, hogy amíg nem alakul ki benne kör, addig egy alkategóriának több szülője is lehetséges. (Azaz egy irányított körmentes gráf, melyben az irányított élek végpontja a szülő, a kezdőpontja a gyermek.)



9. ábra: Egyszerű kategória-fa

A kategória fa éleit súlyértékekkel látjuk el, mely a „kategória-öröklődés” mértékét jelenti. A kifejezés idézőjeles, mert valójában a szülő „örökli” ilyen mértékben a gyermek jellemző tulajdonságait.

5.1.2. Kategóriák szókészlete

A kategóriákat szókészlettel, a szókészleten belül súlyeloszlással jellemezzük. A kategóriára jellemző, fontosabb szavakat nagyobb súllyal számítjuk, mint az átlagos, vagy a nem fontos szavakat. A további néhány bekezdésben a szókészlet alatt nem csak a szavak felsorolását, hanem egymáshoz viszonyított jelentőségüket, súlyaikat is értem, de ezt nem írom ki minden esetben.

Saját és örökölt szókészlet

Kétféle szókészletet különböztethetünk meg származás alapján. Az egyik a kategória saját szókészlete, melyet valamilyen módon beállítottunk. Másik része a szókészletének az „örökölt” szókészlet. Az öröklődés speciálisan a gyermektől a szülő irányába hat, azaz a gyermekkategória szókészletét örökli a szülő, bizonyos tompítással, mindenképpen kisebb értékkel.
(Mind a pozitív, mind a negatív visszacsatolást örökli.)

Pozitívan és negatívan visszacsatolt szókészlet

Visszacsatolás alapján is kétféle szókészlettel dolgozunk. Pozitívan visszacsatoltnak nevezzük azt a szókészletet, mely hagyományos értelemben a kategóriát jellemzi. A szavak a kategóriához tartozó szövegekben gyakran fordulnak elő, s ha egy szöveg sok olyan szót tartalmaz, ami ebben a szókészletben szerepel, akkor bátran sorolhatjuk az adott kategóriához.

Negatívan visszacsatolt szókészlet a téves kategória-besorolásokhoz kapcsolható. Ha egy rendszer a tanulási fázis valamelyik állapotában rossz helyre sorol egy szöveget, s egy adminisztratív személy (például az önkormányzati ügyintéző) ezt jelzi számára, akkor az adott szöveg szavait a negatívan visszacsatolt szókészletbe helyezi.

Az elemzés során először összegezzük a pozitívan visszacsatolt és a pozitívan öröklött szókészleteket, majd levonjuk belőle a negatívan visszacsatolt és öröklött eloszlást. A négy érték végül meghatározza az adott kategóriába tartozás súlymértékét. Mind a negatív, mind a pozitív visszacsatolás külön-külön gyűrűzik végig fel a kategória-fában, így semmilyen fontos információ nem veszhet el.

Tiltott szavak listája

A fenti szókészleten túl definiálhatunk egy olyan szólistát, melynek tagjaival az elemző semmilyen módon nem foglalkozik. Ezek a szavak a tanítás során automatikusan kikerülnek az egyes kategóriákból, elemzéskor pedig az elemzendő szövegben hagyjuk figyelmen kívül őket.

5.1.3. Kategóriába tartozás vizsgálata

Fa-bejárás

Kétféle fa-bejárási algoritmust hoztunk létre. Az első nem is számít algoritmusnak, ugyanis a fa összes elemét megvizsgálja. A második kihasználja a fa hierarchikus tulajdonságát, s megpróbálja modellezni a témamegértés folyamatosan finomodó modelljét. Először csak közelítően határolja be a témát, majd egyre inkább finomítja a részleteket.

A módszer lényege, hogy a kategória-fa gyökereiből kiindulva járja be a fát, egészen addig a pontig, ahol megtalálni véli a legkedvezőbb kategóriát.

- 1. lépés: Megvizsgálja a kategória-fa gyökereit.
- 2. lépés: Kiválasztja az eddig elemzett kategóriák közül a legnagyobb elemzési értékűt.
- 3. lépés: Ha van még nem vizsgált gyermekkategóriája, akkor megvizsgálja azokat, majd a második lépésre ugrik.
- 4. lépés: Ha minden gyermekét megvizsgáltuk már, akkor visszatérünk az adott kategóriával, mint a legnagyobb elemzési értéket elért besorolással.

A szókészletek „öröklésével" (azaz gyökérbe gyűrűzésével), s a fenti folyamatosan mélyülő kategória-elemzéssel egy olyan modellt alkottunk, mely nagyon hasonlít arra, amikor ismeretlen szöveget olvasva megpróbáljuk meghatározni, hogy miről szól. Először csak pl. a tudományterületet tudjuk megmondani (fizika), s csak bizonyos jellemző szavak hatására tudjuk eldönteni, hogy kvantumfizikáról, vagy szupravezetők kialakításáról szól a szöveg.

Számítás

A kategória-elemzés során minden egyes elemzés két számértéket ad, melyek együttesen jellemzik a kategóriába-tartozás mértékét:

- Az ***elemzési bizonyosság*** azt a százalékos értéket jelenti, hogy a vizsgált szöveg szavainak hány százaléka található meg a kategória szókészletében.
- A ***számított értéket*** úgy kapjuk, hogy a kategória és a vizsgát szöveg közös szókészletét a kategória súlyértékei alapján összegezzük.

Mind a szövegek vizsgálatakor, mind a kategóriák elemzésekor csak a fontosabbnak tekintett szavakat vizsgáljuk, névelők, névmások, számnevek nem számítanak, csak a főnevek, melléknevek és igék.

5.1.4. Öntanuló kategória-elemzés

A kategóriák szókészletének kezdeti kialakítása többféleképpen is elvégezhető, de a gyakorlati használat során szükséges a súlykészlet adaptív alkalmazkodása az új szövegekhez és esetleges módosított kategória-struktúrához.

Az elemzőnk felépítése erre is ad lehetőséget, a következő folyamat szerint:

- Az újonnan érkezett szöveget az aktuális szóeloszlás alapján megvizsgálva besoroljuk az egyik kategóriába.
- A rendszer ennek alapján a megfelelő ügyintézőnek továbbítja a levelet.
- Ha az ügyintéző úgy ítéli meg, hogy az a levél nem hozzá tartozik, akkor az adott kategóriára nézve negatívan visszacsatoljuk a szöveget (lásd negatívan visszacsatolt szavak), s egy új kategóriába soroljuk át.
- Ha végül elérkezik a megfelelő ügyintézőhöz, akkor ő azt fogadja, s pozitív visszacsatolás útján tudatja a rendszerrel, hogy a hasonló leveleknek hozzá kell érkezniük.

Rendszerleállás, kritikus helyzet, vagy egyszerű adminisztrátori beavatkozás során az eltárolt szövegekből újra fel lehet építeni a kategóriafát. Ebben az esetben arra is van lehetőség, hogy ne csak a hosszú távú adaptív alkalmazkodásra készítsük fel a rendszert, hanem a szókészlet súlyozásával addig iteráljuk a betanító lépéseket, amíg minden ismert szöveg elemzése a megfelelő kategóriába helyezi azt.

5.2. Állítások

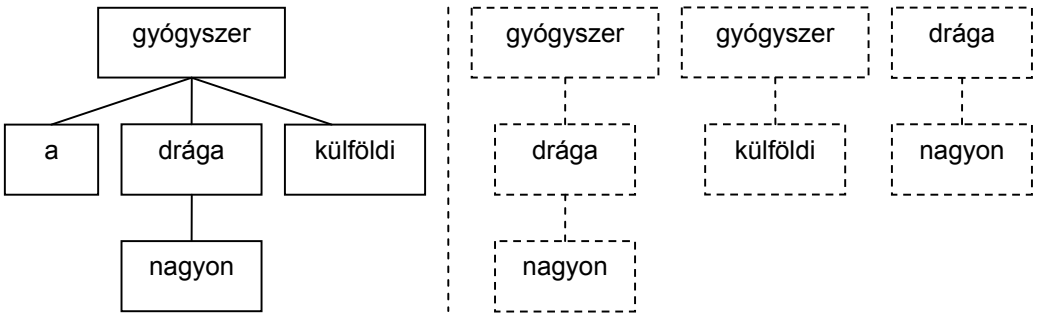
Egy szöveg olvasása során bekezdéseket, mondatokat, mondatrészeket olvasunk. A számunkra érdekes információ ezek közül bármelyik lehet, a szöveg felépítésétől s kíváncsiságunktól függően. Elemző rendszerünkben egy olyan elemi információ egységet jelöltünk ki építőkockának, mely a nyelvtani szerkezetek és a mondatok között helyezkedik el, általában mondatrészként. Az információs közlési egységet állításnak neveztük el.

Állítás lehet például a „Hideg van”, a „Nagy az autóforgalom”, „A zaj zavaró”.

5.2.1. Szerkezeti felépítés

Állítástöredék

Legkisebb építkezési egységünk a nyelvtani elemzés szerkezetéből kerül ki. Állítástöredéknek egy olyan nyelvtani struktúrát nevezünk, mely a nyelvtani elemzési fában egy gyökérből induló úton, nem feltétlen egymás után következő csomópontokon keresztül elérhető.



10. ábra: Nyelvtani elemzés és néhány állítástöredék

A töredékek definiálása röviden:

- A töredék egy vagy több blokkból áll. A blokkok a nyelvtani elemzési fában egy szigorúan mélyülő úton helyezkednek el.

- A töredék felismeréséhez minden blokknak jelen kell lenni a nyelvtani elemzés vizsgált részén belül. A blokkoknak nem kell egymás után (alatt) lenniük, közbeékelődhetnek más tagok is, de a lefele haladó út megléte, s rajtuk a blokkok egyezése kötelező.

- Blokkok egyezésénél megkötéseket tehetünk a szótőre és/vagy az MSD attribútumokra egyaránt. Csak azok a töredékek számítanak érvényesnek, melyek minden a blokkokban szereplő előírásnak megfelelnek (megegyezik a szótő és/vagy illeszkedik az MSD leíró).

  A korlátok közül célszerű valamelyiket mindenképpen megadni. Ha mindkettőt elhagyjuk, akkor elemzési értelemben egy kötelezően beiktatott szabad rést írunk elő az elemzési fában.

- A blokkok a nyelvtani struktúrában nem feltétlen egymás után, nem feltétlen az abszolút gyökérelemből kiindulva helyezkednek el.

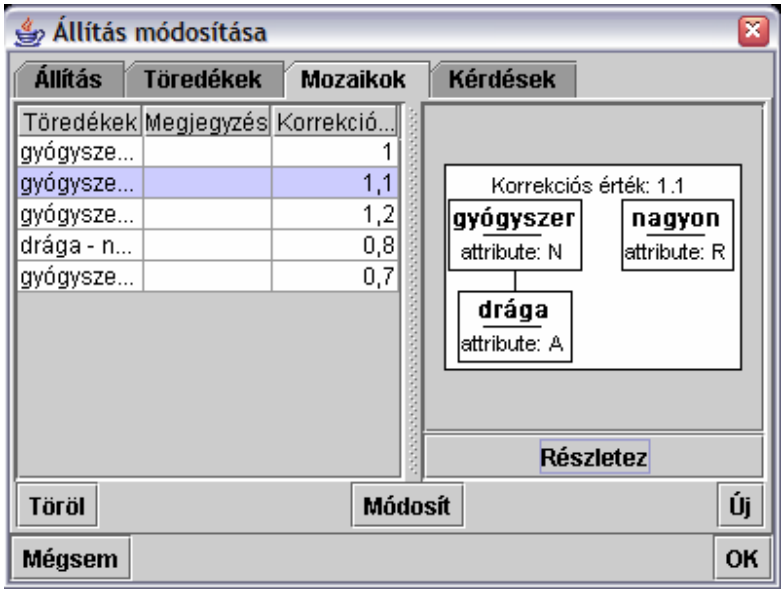
- A töredék felismerési értéke a töredékben szereplő szavak eredetiségi értékeinek a szorzata.

Állításmozaik

Egy állítást akkor mondunk az adott szövegrészen belül bizonyos valószínűséggel megjelenőnek, ha előre meghatározott, együtt előforduló töredékeket találunk. Az adott szövegrészt elemzéseink során mondatokra korlátoztuk, azaz ha egy mondaton belül minden előírt állítástöredék előfordult, akkor az állítást a mondaton belül felismertnek tekintjük.

Az együttesen előforduló állítástöredékeket állításmozaikoknak neveztük el. Egy állításhoz több mozaik is tartozhat, s a mozaikok tetszőleges töredékekre hivatkozhatnak.

Az állítás felismerésének mértéke a mozaikban szereplő töredékek felismerési értékeiből és a mozaikhoz tartozó korrekciós értékből számolt szorzat. Egy szövegrészen (esetünkben mondaton) belül mindig csak a legnagyobb elemzési értékkel rendelkező mozaikot tekintjük, amit az állítás felismerésének nevezünk.



11. ábra: Állításmozaikok adminisztrációs felülete

5.2.2. Interaktív visszakérdezés

A szemantikus elemzés kisebb-nagyobb bizonytalansági tényezővel ismeri fel az állításokat, ezért a rendszer működésének ismeretében nagyon könnyű ügyesen szerkesztett szöveggel becsapni. Ennek ismeretében interaktív mód esetén lehetőséget adunk a rendszer számára, hogy bizonytalanul felismert állításokra visszakérdezzen.

A visszakérdezés a felismerési érték tartományától függ, állításonként beállítható tartományokkal, tartományonként beállítható kérdéssel, s rá adható válaszlehetőségekkel, következményeikkel. Jól összeállított mozaikoknál alacsony és relatív magas elemzési tartományban felesleges a visszakérdezés, ugyanis a mozaikok alapján egyértelműen feltételezhető az állítás hiánya illetve megléte.

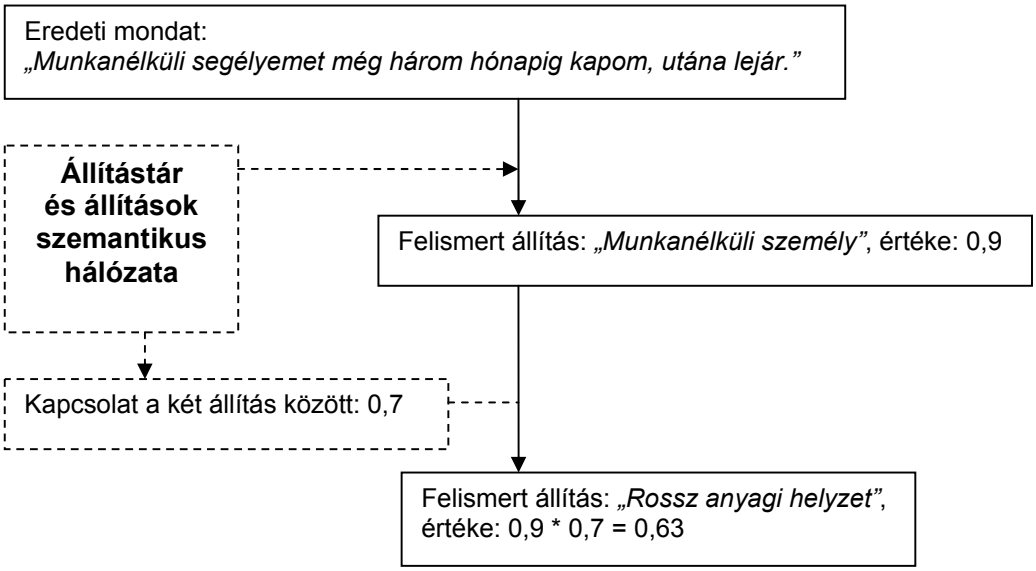
Tipikus visszakérdezés lehet például a következő:

- *„Itt arra gondolt, hogy a szomszéd kutyája megharapta Önt?”*
- *„Itt arra gondolt, hogy fel akarja jelenteni a szomszédját?”*

Az adott szövegrész a felhasználó figyelmének felkeltésére kiemelhető, s a kérdés mellett megjelenő listából kiválaszthatja a számára szimpatikus választ. Az elemző a válasz függvényében módosítja az adott állítás szövegrészen belüli felismerési értékét.

5.2.3. Állítások szemantikus hálózata

Az állítások között definiáltunk egy szemantikus hálózatot. Működésének lényege, hogy a szűkebb jelentéstartalmú állítás maga után vonhat egy hasonló jelentésű, de tágabb állítást.

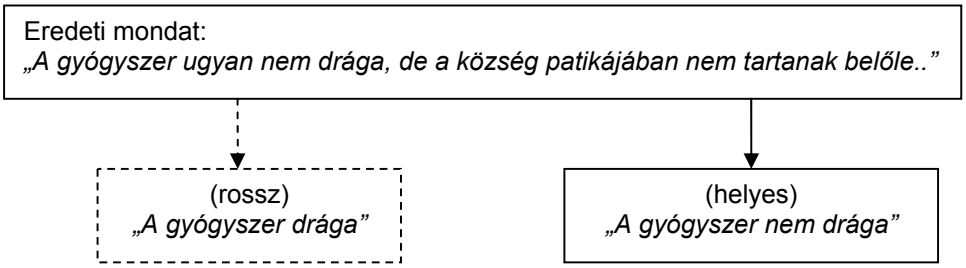


12. ábra: Állítások szemantikus hálózata működés közben

A módszerrel olyan háttértartalom is megjelenik elemzési eredményként, ami nem szerepel közvetlen módon a szövegben, csak természetesnek tartott ismereteink alapján következtetünk rá. Az életben megjelenő tapasztalati tudást tudjuk ilyen módon az elemzésben megjeleníteni.

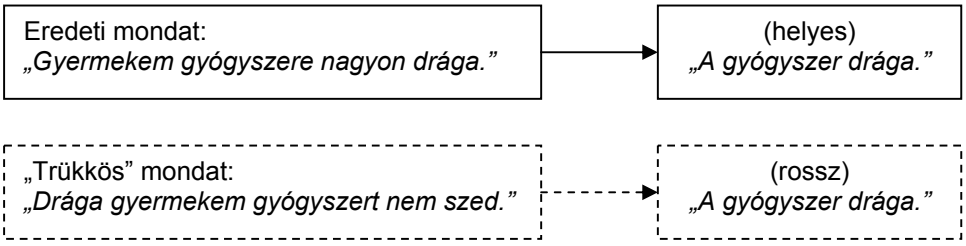
5.2.4. Nem megoldott problémák

A cselekvés, történés, létezés vagy valamilyen állapot fennállásának eseménytere állításokból és tagadásokból áll. Terminológiánkban nagyon pontosan kifejező az állítás megnevezés, tagadásokkal ugyanis csak pontatlanul tudnánk kezelni. Sokszor magukat a tagadó szerkezeteket is pozitív állításokként azonosítja be a rendszer.



13. ábra: Problémák tagadó szerkezetekkel

A nyelvtani elemzés mélységének hiányában nehéz pontosan megfogalmazott töredékeket és mozaikokat írni. Ezen hiányossággal a kezdetekkor is tisztában voltunk, de feltételeztük, hogy egyszerű nyelvtani szerkezetek (birtokos, tárgyas szerkezetek) felismerésével is megközelíthetjük célunkat.



14. ábra: Problémák „trükkös” megfogalmazásokkal

Tapasztalataink minket igazoltak: (nem feltétlen szándékos) trükkös megfogalmazások esetén ugyan nagyobb hibaaránnyal ismerjük fel az állításokat, de az általános szövegek találati eredményei megfelelőek voltak számunkra.

5.3. Forgatókönyvek

Az önkormányzati ügyintézéshez eljutó esetek többsége rutinesetnek számít, hasonló körülmények, hasonló végkifejlettel. Ezeket a hasonló rutin eseményeket kivonatolva forgatókönyveket kapunk: a beadvány, a panasz lényeges, a döntéshez és a következtetésekhez szükséges vázlatát.

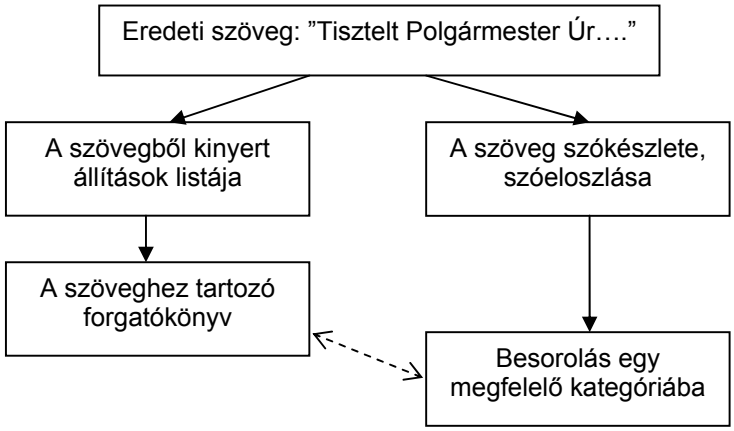
5.3.1. Forgatókönyvek szerkezete

Forgatókönyvön nem a hagyományos értelemben vett folyamatábrát értjük, annál kicsit lazábban értelmezett, kicsit tágabb fogalmat: állítások súlyozott halmazát.

Az elemzés során nem követeljük meg, hogy minden forgatókönyvi állítás szerepeljen a szövegben: elhagyhatóak állítások, de megengedjük más állítások jelenlétét is. Egy-egy forgatókönyv és a szöveg állításainak összehasonlítása során csak a közös állítások súlyozott és normalizált összegére vagyunk kíváncsiak, azaz a forgatókönyvet illesztjük a szövegre, nem fordítva.

5.3.2. Kategóriákra épülő megértési modell

A forgatókönyvet mint fogalmat több hasonló eset absztrakciójaként vezettük be. Azonban amíg nincs elegendő szövegünk, addig a meglévő kevés mintaszövegből kell kialakítani forgatókönyveket. A mintaszöveget az elemző többi részével megvizsgáljuk, s kinyerjük belőle az összes felismerhető állítást. Ez az állításkészlet lesz a szöveghez tartozó forgatókönyv állításhalmaza, melyet operátori felügyelet mellett súlyokkal tovább finomíthatunk.



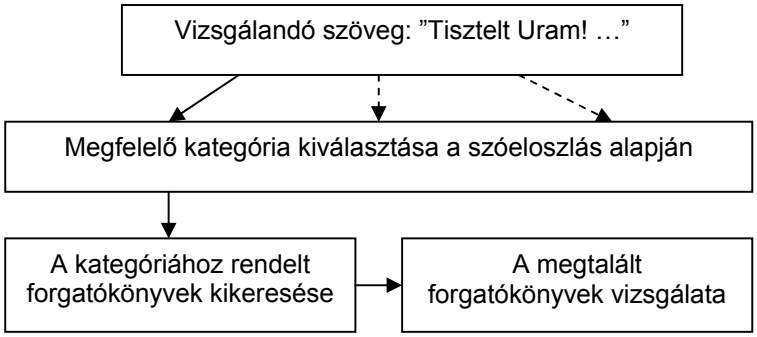
15. ábra: Szöveg, kategória és forgatókönyv kapcsolata

Ha egy forgatókönyv rendelkezik kiindulási szöveggel, akkor jellemezhető a szókészletével is, azaz összekapcsolható a kategória-elemzéssel. A kategória-struktúrába besorolt szöveg forgatókönyvét összekapcsolhatjuk a kategóriával. Ekkor a következő lehetőségeink lesznek az elemző összetett működésénél:

- A kategória szókészletét a hozzá rendelt forgatókönyvek mintaszövegeiből automatikusan képezhetjük. A szókészlet alatt ez esetben nem csak a hagyományos szókészletet érthetjük, ugyanis forgatókönyvek esetén megvan a szabadság arra is, hogy a szövegtől független kulcsszavakkal lássuk el.

- Lefele mélyülő kategória-elemzés után nem szükséges az összes forgatókönyv vizsgálata, elegendő csak a legjellemzőbb kategória részfájába tartozó forgatókönyveket vizsgálni.

Ez utóbbi egy folyamatosan finomodó és pontosító megértési modell alapja, melyet a szűk (állampolgári panaszok és beadványok) tématerületen vizsgáltunk. A kísérleteink során hasonlóan pontos eredményt kaptunk, mint a teljes állítás- és forgatókönyv-gyűjtemény vizsgálatánál, azaz a kategória-elemzéssel összekapcsolva információt nem veszítettünk, ugyanakkor a téma folyamatos szűkítése miatt az elemzés folyamata gyorsabb lett.



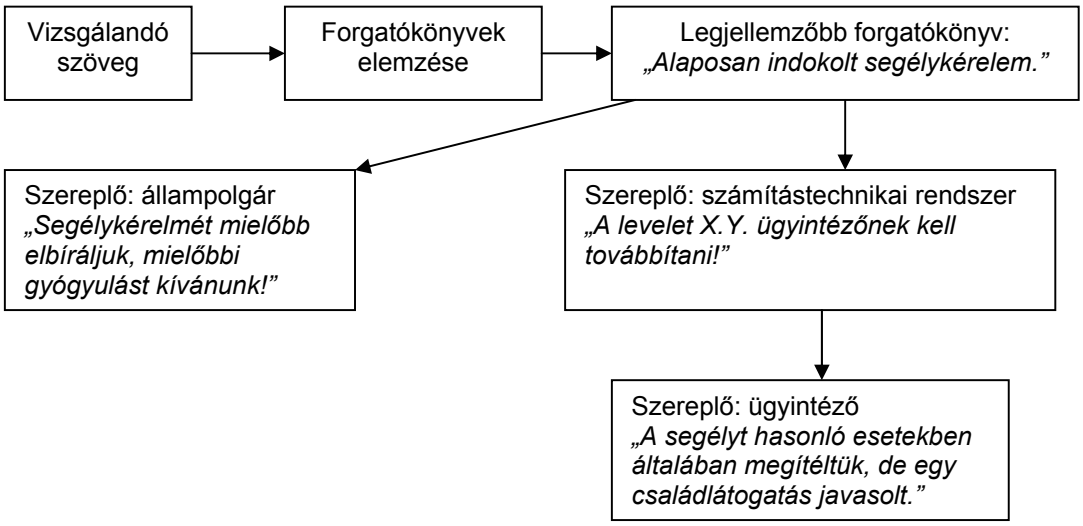
16. ábra: Pontosító megértési modellen alapuló elemzés

Vizsgálataink során nem volt zavaró a szövegkörnyezetek látszólagos hasonlósága, pedig ez lehetett volna a legnagyobb buktatója a statisztikai vizsgálatainknak. Minden tapasztalatunk azt támasztja alá, hogy nem csak ezen a szűk témakörön belül, hanem nagyobb tematikus egységeket feltételezve is hatékonyan működő modellt kaptunk.

5.3.3. Következtetések

A forgatókönyvek eddigi leírása még csak a kezdeti induló feltételek felsorolását és azonosítását jelenti. Ez alapján már meghatározhatjuk, hogy a szövegre mennyire tudjuk az adott forgatókönyvet illeszteni, milyen felismerési értékkel rendelkezik.

Ebből az értékből további következtetéseket vonhatunk le. Például ha a forgatókönyv teljesen illeszkedik a szövegre, akkor az ügyintéző számára tanácsolhatjuk a kért segély megítélését. Ilyen és hasonló esetekre hoztuk létre a forgatókönyvekhez tartozó következtetési rendszerünket. A forgatókönyvhöz szereplőket (actor) rendelhetünk, ami mögött lehet akár automatikus feldolgozó rendszer, lehet a rendszert használó ügyintéző vagy maga a levelet küldő felhasználó.



17. ábra: Következtetésben részt vevő szereplők

A szereplők cselekvései elemzési értékektől függően alakulnak ki. Ha az adott, szereplőnként külön-külön állítható elemzési tartományba belesik az érték, akkor a tartományhoz tartozó utasítás (javaslat, tanács, következtetés…) megjeleníthető, végrehajtható. Ha például egy betegséggel kapcsolatos szöveget küldene a kérelmező, s a forgatókönyv ezt egyértelműen azonosítja, akkor számára megjelenhet a figyelmeztetés, hogy bizonyos tekintetben részletezze a betegségének körülményeit, míg az ügyintéző számára a javaslat, hogy az adott súlyos betegségnél egy gyorssegélyt szinte mindenkinek megítéltek eddig.

5.4. Döntéstámogatás

A korábban említett szemantikai elemzési módszerek eredményeit külön-külön és együtt is felhasználhatjuk számítástechnikával segített döntéstámogatási rendszerekben. A lehetséges felhasználási mód függ a követelményektől és a rendelkezésre álló erőforrásoktól is (az állítások felépítésére még nem dolgoztunk ki automatizmust).

A kategóriákra épülő döntéstámogatási rendszert a kaposvári kísérleti projekt végleges változata is tartalmazza majd. A most következő felhasználási példák is ehhez az önkormányzati feladatokat támogató projekthez kapcsolódnak.

5.4.1. Döntéstámogatás kategóriákkal

A törvények, helyi jogszabályok, az önkormányzat belső felépítése és működésének logikája valamilyen módon struktúrába szervezi azokat az állampolgári beadványokat, melyekkel az önkormányzat kapcsolatba kerül. A döntéshozó rendszerhez ezt a hierarchiát ültetjük át a kategória-elemző struktúrájába.

Ha érkezik egy új elektronikus dokumentum (jellemzően e-mail, vagy webes felületen beérkező kérdés), akkor elolvasásának és értelmezésének feladata nem egy külön ügyintézőre hárul, aki majd továbbítaná azt a megfelelő embereknek, hanem a kategória-elemző vizsgálja meg, hogy milyen jellegű a szöveg, milyen kategóriába illeszkedik leginkább, s a dokumentumot a leginkább illeszkedő kategória illetékeséhez továbbítja.

Az ügyintéző a számára biztosított felületen elolvashatja a beérkezett dokumentumokat, s tevékenységének megfelelően válaszolhat rájuk. Ha úgy látja, hogy az a levél nem a saját feladatköréhez tartozik, akkor negatív visszacsatolással jelzi a kategória-elemzőnek, hogy rossz helyre érkezett a dokumentum. Ha pontosítani akar, akkor új kategóriát is megjelölhet, de ez opcionális. A (passzív vagy aktív) pozitív és negatív visszajelzésekből a rendszer pontosítja a besoroláshoz használt szóeloszlását, így adaptívan alkalmazkodik az új szövegekhez is.

5.4.2. Döntéstámogatás állításokkal

A tapasztalatok alapján jelentős különbségek vannak az egyes panaszosok között. Vannak olyanok, akik tömören leírják, hogy mit szeretnének, de vannak, akik oldalakon keresztül írnak a problémájukhoz közel sem kapcsolódó dolgokról. Az ügyintézőnek ez utóbbi esetben is el kell olvasnia a teljes dokumentumot, mire megtudja, hogy a levélírónak pontosan milyen szándéka is volt a hosszú szöveggel.

Az elolvasást és megértést könnyíti, ha a számítástechnikai rendszer kiemeli, aláhúzással, vastagítással jelzi a felismert, s fontosnak tartott állításokat. Rendszerezett kigyűjtésükkel kivonat is készíthető, így a körülményesen, körmondatokban megfogalmazott panaszok egyszerűbben értelmezhetőek.

Volt néhány kísérleti példaszövegünk, melyekben oldalakat írtak a levél beküldői. Bár az állításaink készlete nem fedte le a szöveg egészét, a felismert állítások gyors elolvasásából jó közelítő képet kaphattunk a levél témájáról.

5.4.3. Döntéstámogatás forgatókönyvekkel

Forgatókönyvek esetében a forgatókönyvhöz rendelt következtetés jelentheti a döntéstámogatás végső állomását. A korábban felsorolt szemantikai elemzések közül a forgatókönyvek elemzése adja a legtöbb információt a szövegről, ezért az ehhez tartozó döntést lehet leginkább személyes, leginkább a témához, a konkrét esethez kapcsolhatóra alakítani.

Az önkormányzati döntéstámogatásban a forgatókönyv következtetésének két jól elkülöníthető felhasználása lehetséges:

- Az állampolgár számára a forgatókönyvtől függő általános tájékoztatást adhatunk, kiegészítő információkat kérhetünk, speciális esetben időpontot foglalhatunk számára.

- Az ügyintéző számára a forgatókönyv egyfajta munkatapasztalatot jelent. Adott esetben javasolhatja a segély elutasítását, figyelmeztethet, hogy helyszíni szemle szükséges vagy – hasonlóan az állampolgári oldalhoz – előhozhatja a törvényi hátterét az ügynek.

6. Az elkészült számítástechnikai rendszer

Kutatómunkánk során több tesztrendszert és programot is készítettünk. Ebben a fejezetben vázlatosan bemutatom a követelményeket és a jelentősebb programokat vagy programrészleteket.

6.1. Számítástechnikai követelmények

Fejlesztői környezet

Fejlesztői környezettel szemben a következő elvárásaink voltak:

- Biztosítson egyszerű hordozhatóságot a különböző platformok (Windows, Linux) között
- Legyen lehetőség egyszerű grafikus megjelenítésre a parancssoros feldolgozás mellett
- Fejlesztői környezete támogassa a gyors és hatékony alkalmazásfejlesztést, vizuális fejlesztői eszközök használatát
- Egyszerűen integrálható legyen webalkalmazásba, web service-környezetbe

A jelenlegi két nagyobb fejlesztői környezetből kellett választani: Java illetve .NET. A Java mellett végül a szélesebb körű ismertsége és támogatottsága miatt döntöttünk.

Moduláris felépítés

Elemző programjaink egy egységes elemző motor köré épülnek. A programok, végezzenek bármilyen szintű elemzési feladatot, ezt a motort egységesen használhatják.

Az elemző motort egymásra épülő rétegekből felépített modulok alkotják. Az egyes modulok kicserélhetőek, esetenként elhagyhatóak, s új modul beillesztése is könnyű feladat.

Szabványos kommunikáció

Fontosnak tartottuk olyan adatszerkezetek kialakítását, amelyek vagy felülről kompatibilisek a korábbi verziókkal, vagy egyszerű transzformációkkal azokká tehetőek. A számítástechnikai kommunikáció nagy része az XML alapú kommunikáció fele mozdult az utóbbi években, így mi is ezt a leíró nyelvet használjuk mind az adatszerkezetek tárolására, mind az elemzési adatok be- és kivitelére.

6.2. Elemző motor

6.2.1. Moduláris felépítés

Elemző környezetünk legfontosabb eleme az elemző motor. Egymásra épülő modulokból áll, melyek egy közös adatszerkezeten dolgoznak. Adatstruktúrája teljesen XML alapú, könnyen illeszthető más rendszerekhez.

A jelenlegi rendszer a következő modulokat tartalmazza:

- Szavakra bontó és rövidítéskezelő modul
- Helyesírás-javító modul
- Morfológiai elemző modul (három részből áll: cache, HuMor és HuMor->MSD átalakító)
- Szinonimakezelő modul
- Nyelvtani elemző modul
- Kategória-elemző modul
- Állításelemző modul
- Forgatókönyv-elemző modul

A modulok működése megfelel az előző fejezetekben tárgyaltaknak.

6.2.2. Közös adatszerkezet

A modulok egy egységesen kialakított adatszerkezeten dolgoznak. Ez az adatszerkezet tartalmaz minden olyan információt, amit az adott modul bemenetként fogad, illetve lehetőséget biztosít az eredmények, az elemzési kimenet tárolására is. A továbbiakban egy feldolgozó program vagy egy további modul feladata az eredmények kiolvasása és tovább értelmezése.

Az adatszerkezet mezői, röviden felsorolva:

- Az elemzendő szöveg (eredeti formában)
- A szavak (darabolt) listája
- Nyelvtani elemzés szerkezeti leírása
- Kategória-elemzés eredménylistája
- Állítás-elemzés eredménylistája
- Forgatókönyv-elemzés eredménylistája

6.2.3. A szöveg és a szavak listája

A feldolgozandó szöveget szövegfájlból és egyéb inputról egyaránt betölthetjük, az elemző szempontjából indifferens. A beállítás során automatikusan figyeljük, hogy tartalmaz-e a szöveg ékezetet, s ha igen, akkor azt egy logikai értéken keresztül jelezhetjük az elemzőnek. Ha nem tartalmaz ékezetet az eredeti szöveg, akkor ez alapján kérhetjük az automatikus ékezetesítést.

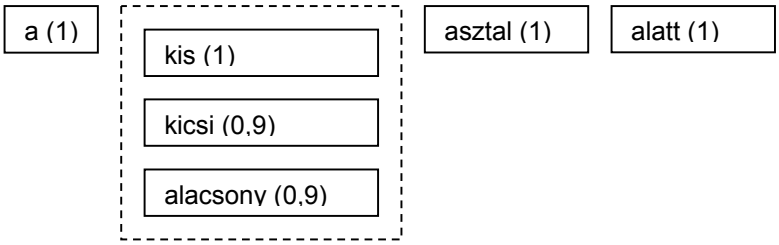
A különböző karakter-kiosztásokból származó kalapos ő és ű betűket is automatikusan a megfelelő formára hozzuk.

Ebből a szövegből képezzük a szavak listáját. A lista minden pozíciójában egy szóhalmaz (s nem csak egy szó) áll. A halmazhoz tartozik az eredeti szó, a szinonimái és ékezetesített alternatívái is. Többféle szűrés is támogatja ezen szóhalmaz méretének csökkentését (például a morfológiailag nem értelmezhető változatok eltávolítása).

A szóhalmaz egy-egy objektuma nem csak az adott szót tartalmazza. A következő információkat tároljuk el minden pozícióhoz:

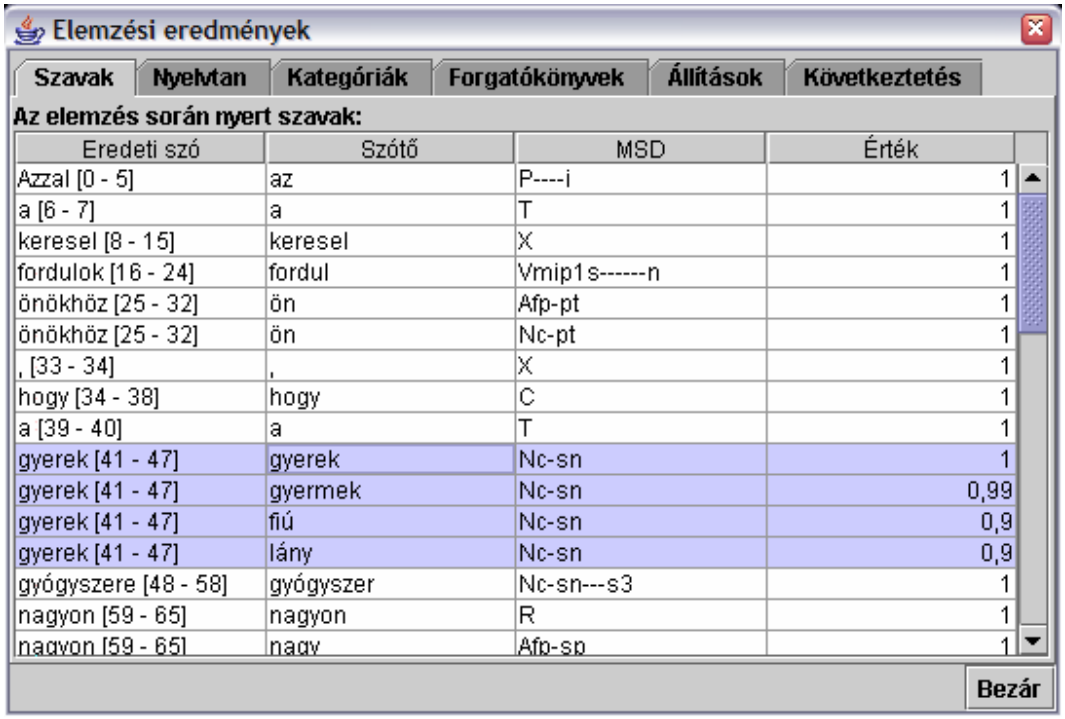
- Az eredeti (származtatási) karaktersorozatot, annak pozícióját
- A származtatási korrekciós értéket (ékezetesítés, szinonimaképzés során egyre kisebb az érték, eredeti szavak esetén 1)
- A morfológiai elemzést, a szótőt és a toldalékok tulajdonságainak leírását.

Egy példa: „a kis asztal alatt” tagolása a következőképpen szemléltethető: A szavak egymás után következnek, de minden egyes pozícióban nem csak egy-egy szó, hanem egy teljes szóhalmaz értelmezendő. A zárójelben lévő szám jelzi az eredetiséget.



18. ábra: Szóhalmazok és eredetiség

A példában a „kis” szót bővítettük a szinonim szókészletével, s így alakult ki a háromtagú szóhalmaz.

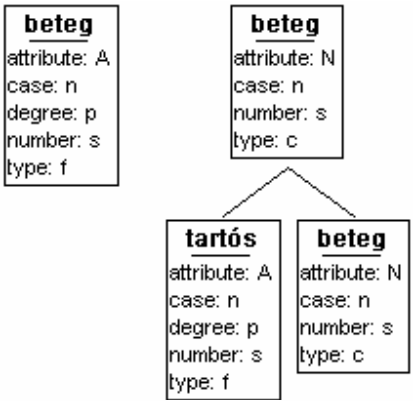


19. ábra: Elemzési eredmény: a szavak listája

6.2.4. Eredménylisták

Nyelvtani elemzés

A nyelvtani elemzés során a különböző hozzárendelési kapcsolatokat egy fa-struktúrában ábrázoljuk. A struktúrarészlet kezdő illetve domináns (pl. főnévi szerkezetnél a főnév) tagjai alapján rendszerezve tudjuk lekérdezni a végeredményt:



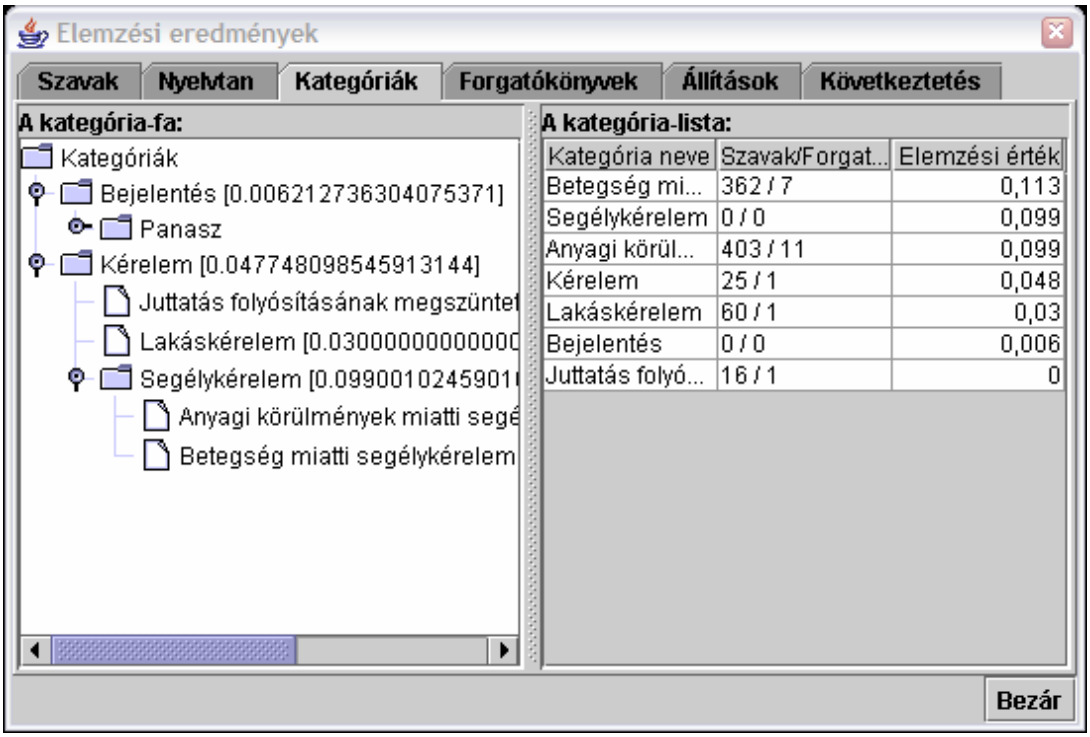
20. ábra: Elemzési eredmény: nyelvtani elemzés

A „tartós beteg” kifejezés elemzése során a beteg szót, mint domináns elemet lekérdezve kapjuk a fenti elemzési képet. Az első fele melléknévként (attribute: A) tartalmazza az egymagában álló beteg szót, míg a második fele egy főnévi

szerkezetben, a számunkra hagyományos struktúrában írja le a „tartós” mint melléknév, s a „beteg” mint főnév kapcsolatát.

Kategóriák elemzése

A kategória-elemzés során minden egyes kategóriához egy-egy számérték kerül. Ezeket egy táblázatból (hash-tábla) olvashatjuk ki:

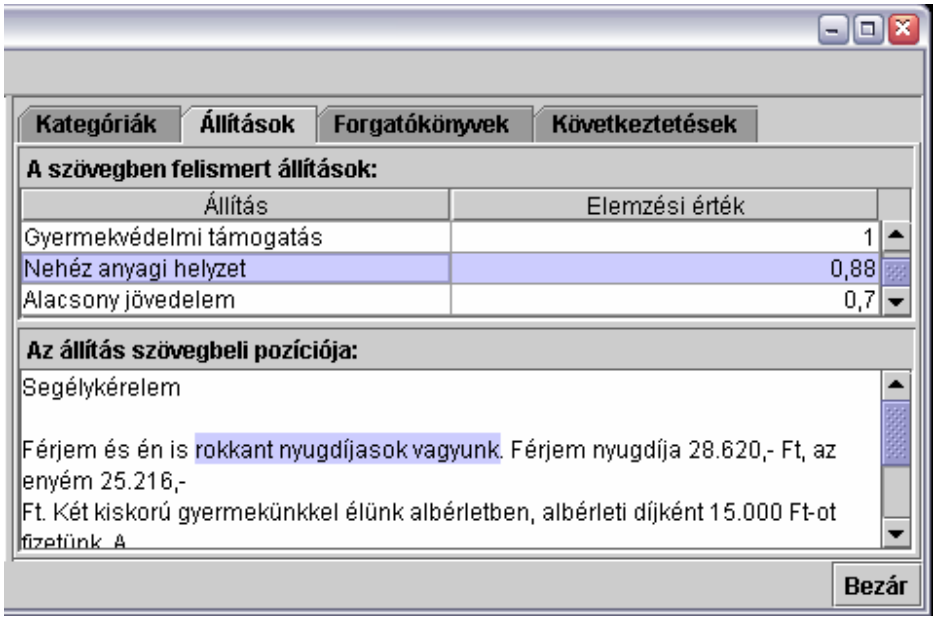


21. ábra: Elemzési eredmény: kategóriák

A fenti ábrán a kategória-elemzés lefele mélyülő elemzési módszerrel történt. Jól látható, hogy a „Bejelentés” nevű főkategória gyermekei nem is lettek kiértékelve, mivel a „Kérelem” nagyobb súllyal szerepelt.

Állítások elemzése

Az állítások elemzése során minden állításhoz eltároljuk, hogy a szövegen belül milyen helyeken milyen értékekkel ismertük fel. Az összes ilyen felismerési hely visszakereshető, azok is, amelyeknél csak az állítások szemantikus hálója alapján (szinonim vagy következmény-állításként) ismertük fel az állítást.

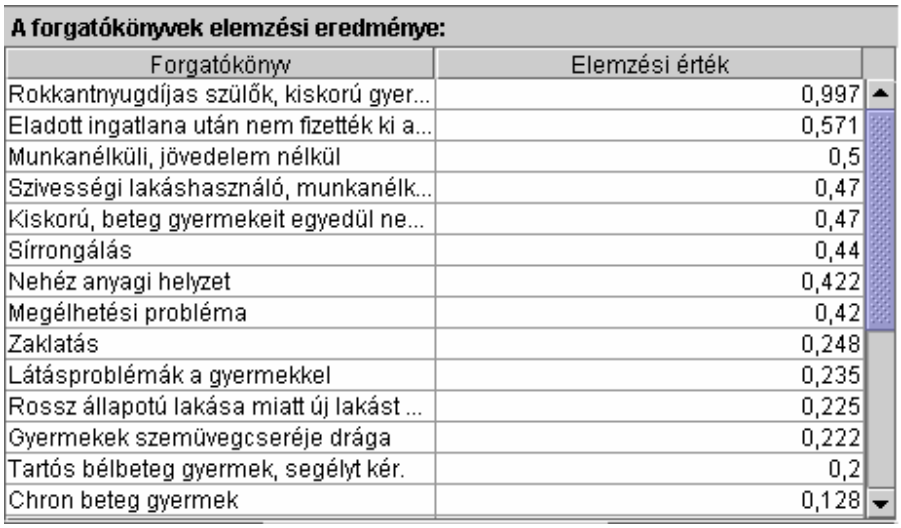


22. ábra: Elemzési eredmény: állítás (szemantikus háló alapján)

A fenti ábrán is egy ilyen szinonim felismerés látható. A szövegből első lépésben a „rokkant nyugdíjas" állítást ismerte fel a rendszer, de a szemantikus háló alapján felismertnek tekintette a „nehéz anyagi helyzet" állítást is.

Forgatókönyvek elemzése

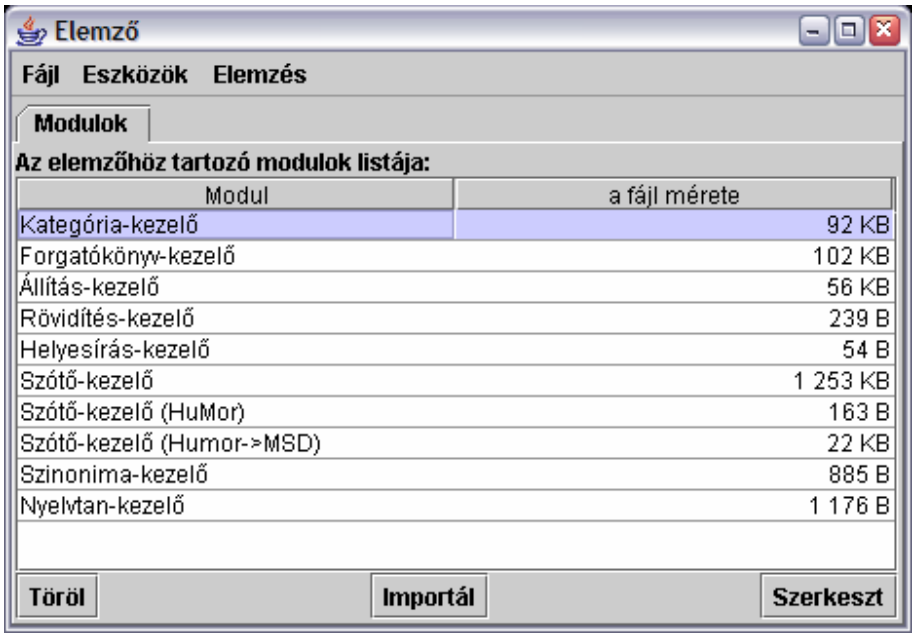
A forgatókönyvek elemzése során minden egyes forgatókönyvhöz egy-egy számérték tartozik. Ezeket egy táblázatból (hash-tábla) olvashatjuk ki:



23. ábra: Elemzési eredmény: forgatókönyvek

6.3. A kísérleti elemző

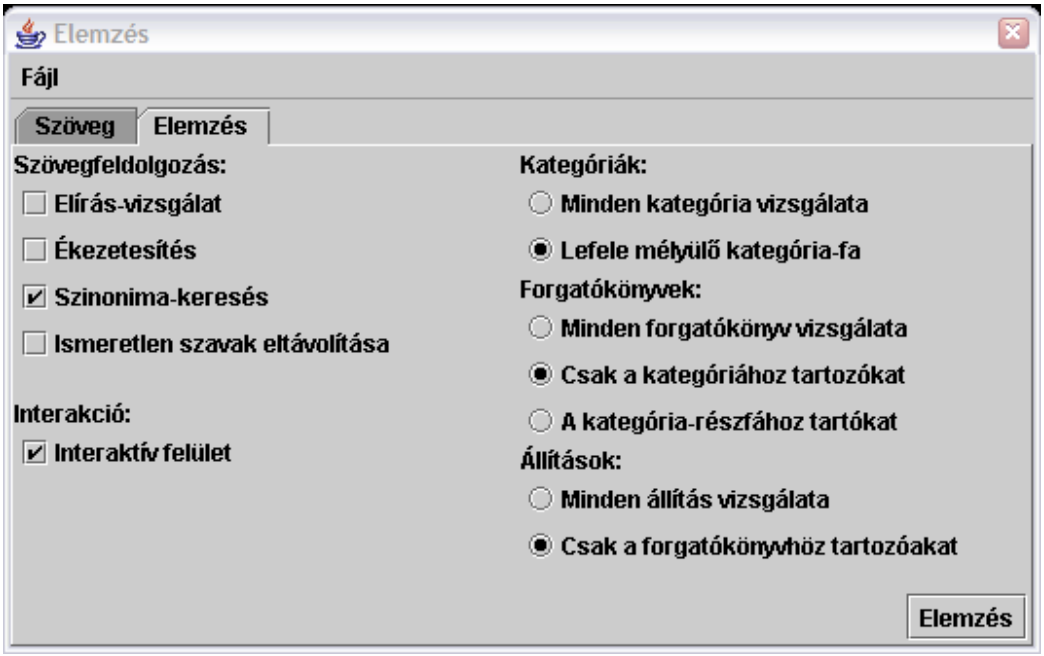
Az elemzési motor teszteléséhez, a szabályok és összefüggések egyszerű adminisztrációjához létrehoztunk egy könnyen bővíthető, közel teljes adminisztrációs felületet.



24. ábra: A kísérleti elemző moduljai

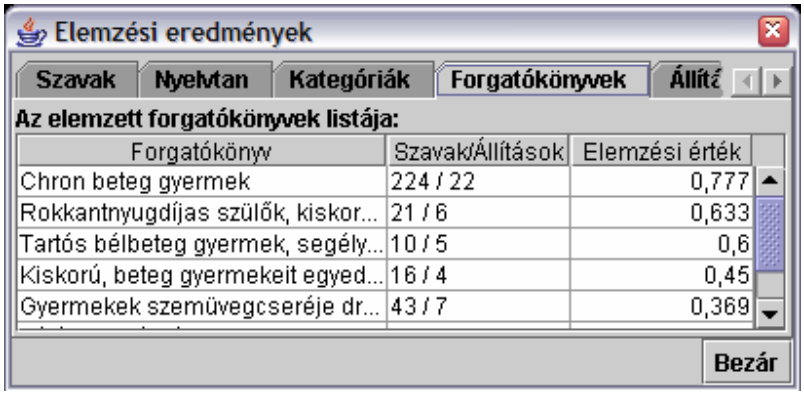
A program lehetőséget biztosít az egyes modulok cseréjére, törlésére, s szerkesztésére egyaránt. A szabályok kezelésén túl képes mintaelemzések futtatására is. Minden egyes elemzésrész eredményét részletes formában is megnézhetjük.

A következő ábrán egy ilyen elemzés beállítási panelje látható. Az opcionális modulokat egy jelölőnégyzet segítségével állíthatjuk, s a szemantikus vizsgálatokra is több elemzési lehetőségünk van.



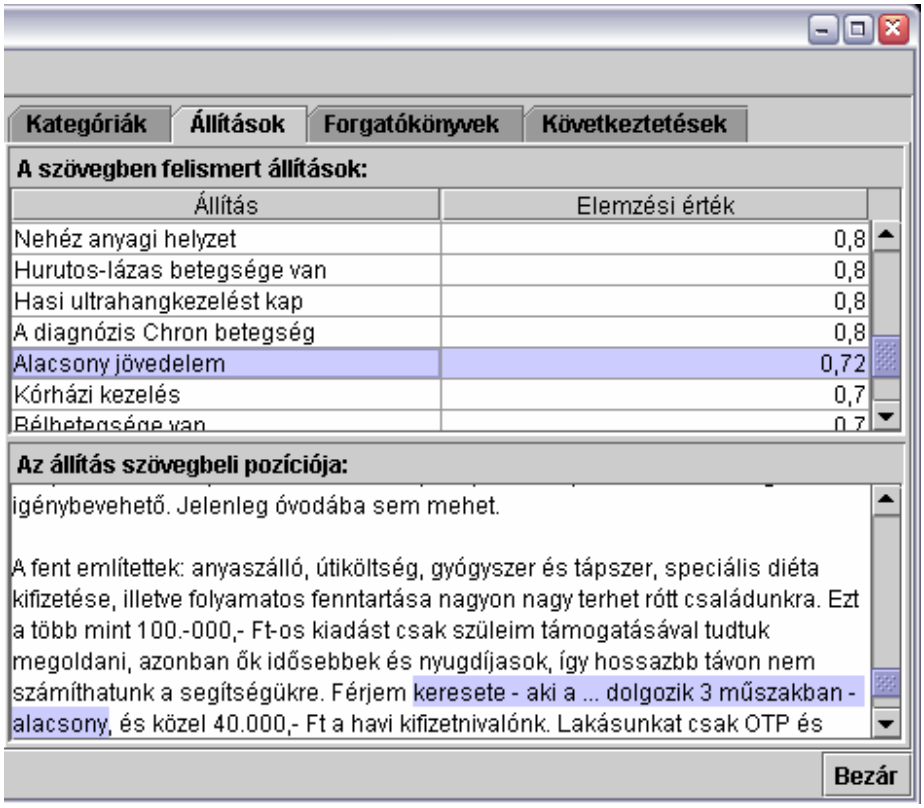
25. ábra: Az elemzés tulajdonságainak beállítása

Az elemzési eredményeket minden modulhoz vagy modulcsoporthoz (a szavak elemzését, a szövegfeldolgozást és az alapvető szintaktikai elemzéseket összevontuk) megnézhetjük:



26. ábra: Elemzési eredmények részletezése

A programnak készült egy speciális demo változata, melyet azért készítettünk, hogy a kaposvári önkormányzat munkatársai egy barátságos felületen tudják kipróbálni és tesztelni az elemzéseket. Az előző programmal ellentétben ezzel csak különböző elemzésekkel lehet kísérletezni, szabályokat nem lehet szerkeszteni. Az eredményeket jóval közérthetőbb formában, de kisebb részletességgel mutatja be.



27. ábra: Az elemző speciális változata vizsgálat közben

6.4. Öntanuló kategória-elemző

Szemantikus elemzéseink közül a végleges kaposvári projekthez csak a leginkább automatizálható kategória-elemzőt használtuk fel. A készülő portálon belül létrehoztunk egy elkülönített részt a tanuló és elemző rendszernek, mely SOAP felületen keresztül kommunikál a portál többi részével. Így web-szolgáltatás kezeli le a beérkező szöveg kategóriájának meghatározását, az adaptív tanításokat, s a szolgáltatással kapcsolatos adminisztratív teendőket egyaránt.

A rendszer megvalósításához Axis-t [6] használtunk. Az Axis-t az Apache csoport részeként fejlesztik, s célja, hogy a programozó számára átlátszó SOAP felületet biztosítson, s a web-szolgáltatások írása egyszerű legyen.

7. Továbbfejlesztési lehetőségek

Elemző rendszerünket több nyelvészeti kutatással foglalkozó csoportnak is bemutattuk, s a pozitív visszajelzéseik mellett továbbfejlesztési javaslataikat is gyűjtöttük. Jelenleg a következő területeken látunk lehetőséget a rendszer aktív továbbfejlesztésére:

Automatizálás

Az állítások létrehozása jelenleg manuálisan történik. A szemantikus elemzésen belül a legnagyobb előrehaladás ezen állítások automatikus vagy félig automatikus félig felügyelt bővítése lenne. Az elemzés tulajdonságainak megőrzése mellett a minél nagyobb fokú automatikus minden területen hasznos.

Állítások tagadása

Az állítások leírásánál kiemeltem, hogy a tagadó nyelvtani szerkezeteket jelenleg nem tudjuk a várt módon kezelni, sőt, általában a fordított állítást ismerjük csak fel. Mélyebb nyelvtani elemzés és az állítástöredékek szerkezetének módosítása esetén ezeket is le tudnánk kezelni.

Állítások következtetési hálózata – új szerkezetű forgatókönyvek

Jelenleg a forgatókönyvek – a név jelentésével kicsit ellentmondásosan – csak állítások halmazából állnak. Ha időben szeretnénk modellezni ezt, akkor folyamat jellegű, ok-okozat összefüggéseket tartalmazó forgatókönyveket kellene készíteni. Ezek sokkal realisztikusabb döntési hálózatot alkotnának, s lehetőség adódik állításrendszerek ellentmondásosságának vizsgálatára is.

Szerző jellemzése

Érdekes terület a hasonló beadványt író emberek jellemzése a fogalmazásmód alapján. Személyre, stílusra jellemző szavak, kifejezések vizsgálatával általános sztereotípiák segítségével tovább finomíthatjuk a rendszer által adható információ mennyiségét.

Zárszó

Az utóbbi évekig kevés alkalmazás követelte meg a szöveg részletes szemantikai elemzését. Professzionális fordító rendszereket leszámítva a legtöbb alkalmazás megelégedett statisztikai eloszlásokkal, kulcsszavak, kifejezések gyűjtésével. A szintaktikai elemzéseket hátráltatja a szövegek nyelvtani feldolgozásának nehézsége, a magyar nyelvre készített nyelvtani szabályrendszer hiánya.

Szemantikai elemzésünket úgy építettük fel, hogy a jelenleg könnyen elérhető szintaktikai struktúrákat a lehető leghatékonyabban felhasználva a legtöbb információt kinyerjük a szövegből. Az elektronikusan segített önkormányzati rendszer kiváló kísérleti terepnek bizonyult: rengeteg tapasztalatot és megvalósítási ötletet nyertünk a felmerülő problémákból, s a megoldási kísérleteinkből.

Automatizálási programjaink végső célja az emberi munka megkönnyítése, egyszerűsítése, a felesleges időtöltések kiszűrése. Ezeket a feladatok előtérbe helyezve készítettük el összetett rendszerünket, melyet a döntéstámogatás több pontján is hatékonyan fel lehet használni.

„A” melléklet: Elemzési példa részletezése

A mellékletben végigkövetünk egy valódi szöveget, s a legtöbb elemzési eredményt részletesen bemutatom rajta. A szöveg elektronikus levél formájában érkezik, s az összes modult belevonom a feldolgozásba.

A.1. Az elemzendő szöveg

A példában szereplő szöveg valódi beadvány valósághű, javítás nélküli számítógépes átirata:

```
Tisztelt Gyámhatóság!

Három iskolás gyermekem közül kettő állandó szemüveges. Nagy gondot
jelent, mikor egyszerre kell szemüveget csináltatni vagy éppen eltörik.
Most sajnos még itt volt az iskolakezdés is. GYES-en vagyok, férjem az
egyedüli kereső, így mindig nem bírja a családi kassza, ha muszáj is
lenne, az ilyen kiadást.
Nikolett szemüvegét tudtuk csak megcsináltatni, pedig Gábornak is
sürgősen szükséges lenne.
Igaz, kapunk családi pótlék kiegészítő támogatást, de az elmegy az
iskolához. Kérem, ha lehetséges, rendkívüli segélyt részünkre
megállapítani, a TB már egyáltalán nem támogatja a gyermekszemüvegeket.
A családban mi, szülők is szemüvegesek vagyunk, sehonnan semmi
támogatást erre nem kapunk.
Kérjük, méltányolják helyzetünket!
```

A.2. Elemzési beállítások

Az elemzés során az összes elkészült modult felhasználjuk. A helyesírás-ellenőrző és a szinonimaszótár szinte elhanyagolhatóan kicsi adatbázissal rendelkezik, a nyelvtani elemző pedig csak ötféle egyszerű nyelvtani szerkezetet ismer fel. (Mint később látni fogjuk: a kis szabálybázis ellenére az ezekre épülő szemantikai elemző modulok hatékonyan tudnak működni.)

A szemantikai elemzést a korábban már említett közelítő megértési modellhez alakítjuk, kis kiegészítéssel:

- A kategóriák elemzése folyamatosan mélyülő kategória-struktúra alapján történik.
- Csak a legjellemzőbb kategóriához tartozó forgatókönyveket vizsgáljuk.
- Bár nem lenne szükséges, az összes állítást megvizsgáljuk, nem csak a forgatókönyvek szempontjából kritikusakat.

A.3. Szavak elemzése

Terjedelmi okokból nem sorolom fel a szövegben előforduló összes szó elemzését, csak az első néhányat:

Eredeti szó	Szótő	MSD kód	Eredetiség mértéke
Tisztelt [0 – 7]	tisztelt	Afp-sn	1.0
Tisztelt [0 – 7]	tisztel	Vmis3s------n	1.0
Gyámhatóság [9 – 19]	gyámhatóság	Nc-sn	1.0
Gyámhatóság [9 – 19]	gyámhatóság	Nc-sn	1.0
! [21 – 21]	!	X	1.0
Három [23 – 27]	három	Mc-snl	1.0
iskolás [29 – 35]	iskolás	Afp-sn	1.0
gyermekem [37 – 45]	gyermek	Nc-sn---s1	1.0
gyermekem [37 – 45]	gyerek	Nc-sn---s1	0.99
gyermekem [37 – 45]	fiú	Nc-sn---s1	0.9
gyermekem [37 – 45]	lány	Nc-sn---s1	0.9
közül [47 – 51]	közül	R	1.0
közül [47 – 51]	köz	Nc-sw	1.0
közül [47 – 51]	köz	Afp-sw	1.0
kettő [53 – 57]	kettő	Mc-snl	1.0
állandó [59 – 65]	állandó	Afp-sn	1.0
állandó [59 – 65]	állandó	Nc-sn	1.0
szemüveges [67 – 76]	szemüveges	Afp-sn	1.0
. [78 – 78]	.	X	1.0

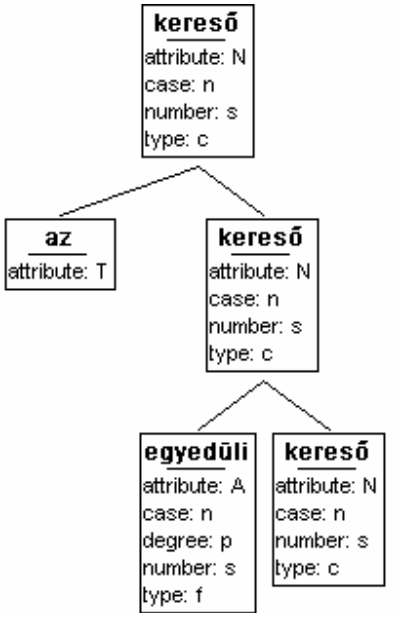
Az első oszlop az eredeti szót, s annak az eredeti szövegben elfoglalt pozícióhatárait tartalmazza. Ez az információ a szöveg darabolásától kezdve a szó tulajdonsága, a későbbi összes szerkezetnél visszakérdezhető.

A második és harmadik oszlop a szó morfológiai elemzésének eredményét tartalmazza. A szótő és az MSD kód együttesen jellemzik az eredeti szóalakot. Az „X” jelölésű MSD kód fel nem ismert szerkezetet takar.

Az utolsó oszlop a szó adott változatának eredetiségi mutatója. Jól látható, hogy a példában csak a szinonim szóalakok eredetisége csökken, a nyelvtani elemzés különböző változatai eredetinek számítanak a későbbiekben is, hiszen az elemzés ezen pontján még nem ismerjük, hogy melyik a helyes alak.

A.4. Nyelvtani elemzés

A nyelvtani elemzés során több elemzési részfát is kapunk. A hely és a formátum miatt az elemzési eredményeket is a teljességre való törekedés nélkül mutatom be, néhány egyszerűbb példán keresztül:



28. ábra: Nyelvtani elemzés: „az egyedüli kereső”

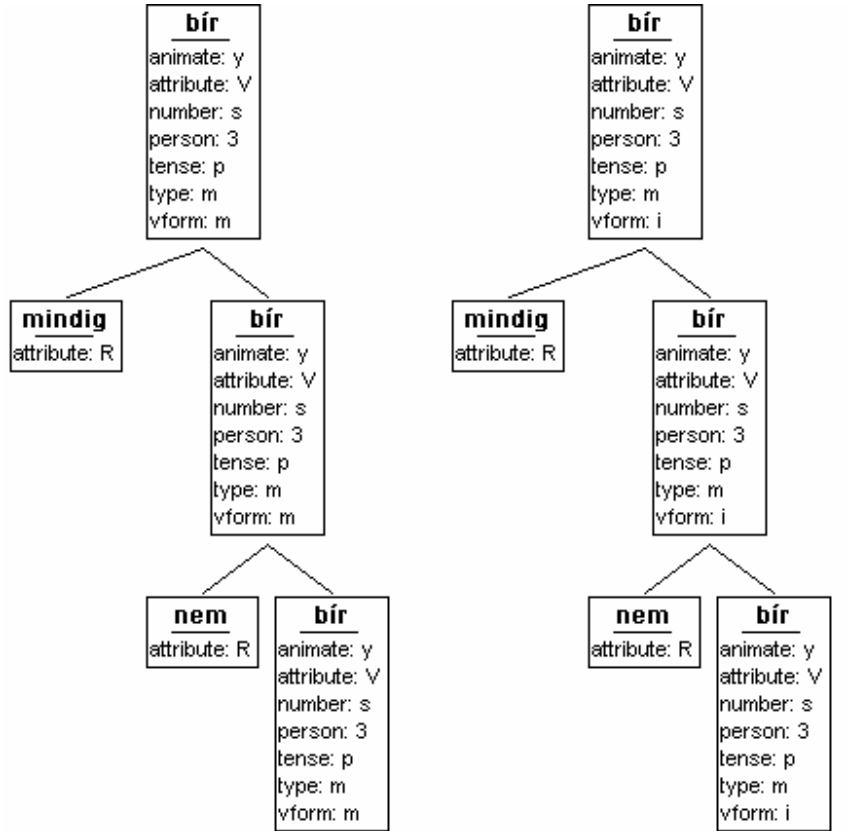
Az összevont nyelvtani szerkezet „az egyedüli kereső” szövegrész elemzése. Az elemző első lépésben összevonta a „kereső” (főnév) és az „egyedüli” (melléknév) szót, s a „kereső” szót jelölte meg a szerkezet fő elemének. További MSD attribútumot nem adott a szóhoz, megőrizte az összes tulajdonságát.

Második lépésben a szerkezet előtt álló „az” névelővel vonta össze az eddig elkészült szerkezet (főnév szófajú) főelemét (s nem az eredeti „kereső” szót, bár ez esetben módosítók hiányában nincs lényeges különbség).

A következő példa a „mindig nem bírja” szerkezet lefedési elemzését mutatja be. A szövegkörnyezet a következő volt: „így mindig nem bírja a családi kassza”, de ebből csak az előbb említett három szót tudtuk csak lefedni nyelvtani szerkezettel.

Az ábra felépítése hasonlít az előzőhöz, hiszen először a „bír” igét vonja össze a „nem” határozószóval, majd a „mindig” szót az előbb nyert szerkezet főelemével, azaz a „bír” igével.

Az előző ábrától egy lényeges eltérés különbözteti meg: mivel a „bírja” ige kétféle elemzéssel rendelkezik, a szavak listájában is két (különböző) elemzési példánya jelenik meg. A kettő között az eltérés a kijelentő és felszólító módú igealak között jelentkezik, de ez a kis különbség is megköveteli, hogy a két igealakra külön-külön felépítsük az amúgy nagyon hasonló elemzési részfát.

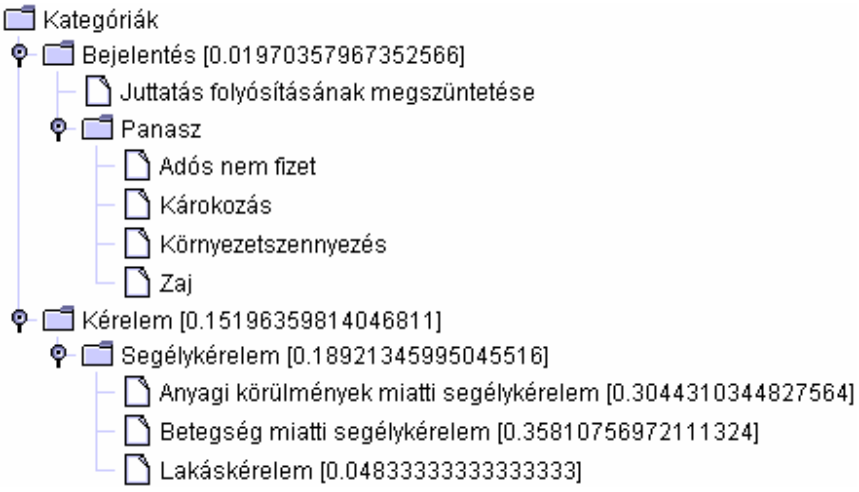


29. ábra: Nyelvtani elemzés: „mindig nem bírja”

A fentiekhez hasonló szerkezetekből épül fel a teljes nyelvtani elemzési erdő.

A.5. Kategóriák elemzése

A kategóriákat a következő struktúrába szerveztük (zárójelben az adott kategória elemzési értéke)



30. ábra: Kategóriák elemzése: folyamatosan mélyülő módszer

Az ábráról jól leolvasható a módszer működése: először a „Bejelentés” és a „Kérelem” közül kellett választania, majd folyamatosan mélyülve eljutott a „Segélykérelemig”, azon belül a „Betegség miatti segélykérelemig”.

Kategória megnevezése	Elemzési eredmény
Betegség miatti segélykérelem	0.358
Anyagi körülmények miatti segélykérelem	0.304
Segélykérelem	0.189
Kérelem	0.151
Lakáskérelem	0.048
Bejelentés	0.019

A kategória-adatokat csökkenő sorba rendezve az láthatjuk, hogy a domináns elemzési értékek közül a betegség és az anyagi körülmények miatti segélykérelem kategóriák nagyon hasonló elemzési értéket adtak. Nem véletlenül: ha visszaolvassuk a szöveget, akkor érezzük, hogy bizony betegség miatt és anyagi körülmények miatt is kéri a segélyt az illető édesanya.

A.6. Állítások elemzése

Az állítástárból a következő állításokat ismerte fel a rendszer:

Állítás megnevezése	Elemzési eredmény
Segélykérelem	1.0
GYES	1.0
Szemüveges	1.0
TB nem támogatja a gyermekszemüveget	1.0
Látászavar	0.9
Szemüvegviselés indokolt	0.7
Nehéz anyagi helyzet	0.7

Az első három állítás általános módon megfogalmazott állítás, nem csak ebben a szövegben fordult elő. A negyedik állítást a példaszöveg elolvasása után hoztuk létre, ugyanis korábbi szövegeinkben ilyen nem fordult elő. Az utolsó három állítás az állítások szemantikai hálózata alapján került bele a szövegbe, közvetlenül sehol sincs leírva, de jelentéstartalmuk érezhető.

A.7. Forgatókönyvek elemzése

A rendszer a „Betegség miatti segélykérelem” kategóriához tartozó forgatókönyveket elemezte. A kategóriában nyolc darab szöveget (illetve a hozzájuk tartozó forgatókönyvet) dolgoztuk fel eddig, az elemzési eredményeik a következők:

Forgatókönyv megnevezése	Elemzési érték
Gyermekek szemüvegcseréje drága	0.9153
Kiskorú, beteg gyermekeit egyedül neveli	0.4250
Látásproblémák a gyermekkel	0.4125
Rokkantnyugdíjas szülők, kiskorú gyermekekkel	0.2833
Tartós bélbeteg gyermek, segélyt kér.	0.2000
Gyerekek hosszu ideje betegek	0.1761
Chron beteg gyermek	0.1119

Az első forgatókönyv kiemelkedően magas felismerési értékkel rendelkezik. Ez nem véletlen, ugyanis ezek a forgatókönyvek nem az „ideális" esetet leíró általánosított forgatókönyvek, hanem konkrét esetekből állítás-kiemeléssel létrehozott változatok. A példaként említett szöveget is feldolgoztuk ilyen módon, s az ebből nyert forgatókönyv kapta a legnagyobb elemzési értéket.

Ha az első speciális példát nem vizsgáljuk, akkor szignifikánsan két további forgatókönyv szerepel még relatív magas felismerési értékekkel. Mindkettő hasonló szövegből építkezik. Röviden összefoglalom a „Látásproblémák a gyermekkel" című forgatókönyv jellemzőit:

Az eredeti szöveg a következő volt:

```
....alatti lakos az alábbiakban felsorolt indokok alapján kérem a
Tisztelt Címzett lehetőség szerinti anyagi támogatását.
Több mint két esztendeje egyedül nevelem három gyermekemet: Sarolta és
Lilla ikerlányaimat, 5 és fél esztendősek, Miklós 4 esztendős. Mint
főállású szülő gondoskodom gyermekeimről. Mivel lányaim koraszülöttek,
Lillánál keresztirányú kancsalság és látászavar alakult ki, így állandó
szemüvegviselés indokolt nála. Nemrégiben műtéten esett át, új
szemüveget írt fel az orvos.
Sarolta lányom óvodai szűrés alkalmával arról tett tanubizonyságot,
hogy nála is szemüvegviselés indokolt, ezen felül szemtapasztást is
előírtak szem- és látáserősítés végett. A GYED összege nem elengedő a
gyerekek eltartásához, ezért a Gyámügyi Iroda segítségével a
gyermekeket kollégiumi elhelyezésben, megnyugtatóan sikerült rendezni,
miáltal én munkába tudok állni. A két gyermek szemüvegének ára és az
ünnepi kiadások meghaladták anyagi erőforrásaiomat. Továbbá a
kollégiumi elhelyezés is most nekem megterhelő volt, tartalékaim
kimerültek, ezért kérem a Tisztelt Címzettet, hogy anyagi, rendkívüli
gyermekvédelmi támogatást nyújtani szíveskedjenek! Kérelmemhez
benyújtom a szükséges igazolásokat és számlákat.
```

Ebből a következő állításokat kiemelve készítettük el a forgatókönyvünket:

- Gyermeket egyedül nevel
- Keresztirányú kancsalság
- Látászavar
- Szemüvegviselés indokolt
- Koraszülött
- Segélykérelem

- Műtétet végeztek
- Nehéz anyagi helyzet

Látható, hogy a betegség főbb tünetei, s a segélykérelem ténye közös mindkét forgatókönyvben, s csak speciális, esetre jellemző állításrészekben térnek el egymástól.

A.8. Elemzés értékelése

A kategóriák elemzése során az elemző két nagyon közeli kategóriát jelölt meg a szöveghez. Egyikkel sem tévedett sokat, ugyanis a beadvány tényleg mindkét kategóriába egyformán beleillett. A betegséggel kapcsolatos kifejezések miatt mégis a „Betegség miatti segélykérelem" kategóriába sorolta a program, ami számunkra egy kicsit kedvezőbb találatot jelent. (A példában szereplő elemző környezet betanítása során csak minimális számú olyan szöveg volt, amelyik csak az egyik vagy csak a másik kategóriába tartozott volna, a szövegek többsége ugyanis egyszerre tartozott mindkét segélykérelemhez.)

Az állítások kivonatolása során látható, hogy ez a feladat a rendszer kritikus pontját képezi: gazdag szemantikus tudástárat kaphatunk egy helyesen felépített adatbázisból, de mindig lesznek olyan új szövegek és olyan mondatok, melyből nem tudunk érdemleges információt kinyerni. A szemantikus szövegháló némileg ugyan befoltozza ezt a rést, de nem helyettesítheti teljesen a részletes szemantikai (és nyelvtani) vizsgálatokat.

A forgatókönyvek elemzésénél is éles kontrasztot láthatunk: egy hasonlóan felépített forgatókönyv (vagy egy egyelőre csak elméleti síkon létező általánosított forgatókönyv) nagyon nagy találati pontosságot ér el. Egy logikusan felépített, részletesen kidolgozott rendszer esetén valóban alkalmas a döntéstámogatásra.
Mivel jelenleg a forgatókönyvek konkrét esetek köré épülnek, a találati érték alacsonyabb, de a legjellemzőbbek még mindig hasonló eseteket fognak össze. Sajnos kevés szöveget tudtunk eddig feldolgozni, s még nem dolgoztunk ki hatékony automatizmust a különböző jellegű szemantikai adatbázisaink fél-automatikus bővítésére.

A rendszer korlátain belül az elemzést nagyon jónak tekintjük. Nem csak a rendszer lehetőségei látszódnak az eredmények értékelésekor, de azok a pontok is kirajzolódnak, melyek mentén a továbbfejlesztést elvégezhetjük.

Ábrajegyzék

Irodalomjegyzék

[1] http://www.sztaki.hu/
[2] http://www.morphologic.hu/
[3] http://www.unicode.org/
[4] http://nl.ijs.si/ME/V2/msd/html/msd.html
[5] É. Kiss Katalin, Kiefer Ferenc, Siptár Péter: Új magyar nyelvtan, Osiris, Budapest, 1999
[6] http://ws.apache.org/axis/