Egy pszicholingvisztikai indíttatású számítógépes nyelvfeldolgozási modell felé

Prószéky Gábor[1,2,3], Indig Balázs[2], Miháltz Márton[1], Sass Bálint[1]

[1] MTA–PPKE Magyar Nyelvtechnológiai Kutatócsoport
[2] Pázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar
[3] MorphoLogic
e-mail: {proszeky.gabor, indig.balazs, mihaltz.marton, sass.balint}@itk.ppke.hu

Kivonat Cikkünkben egy, az eddigi megközelítésektől jelentősen eltérő nyelvelemző rendszert ismertetünk, mely a következő alapelvek szem előtt tartásával készül. (1) A *pszicholingvisztikai* indíttatás azt jelenti, hogy amennyire csak lehetséges, az emberi nyelvfeldolgozás mintájára alakítjuk ki a modellt. (2) *Performanciaalapú* rendszerként minden olyan nyelvi megnyilatkozást megpróbálunk feldolgozni, ami (leírt szövegekben) előfordul, nem helyezünk hangsúlyt az elméletileg létező, de a gyakorlatban meglehetősen ritka jelenségek kezelésére. Ugyanakkor bármilyen – rosszul formált, agrammatikus – szöveget igyekszünk nyelvi megnyilvánulásnak tekinteni és értelmezni. (3) Szigorúan *balról jobbra*, szavanként dolgozzuk fel a szöveget. A még el nem hangzott elemeket teljes mértékben ismeretlennek tekintjük, rájuk nem hivatkozunk. (4) Az elemző architektúrája eredendően *párhuzamos*. A hagyományos megközelítéssel szemben, ahol az elemzések egy láncot alkotó modulsor végén alakulnak ki, itt az éppen elemzendő szót folyamatosan, párhuzamosan jelen lévő szálak (morfológiai elemző, korpuszgyakorisági szál stb.) egyszerre vizsgálják és együttesen, egymással kommunikálva, egymás hibáit javítva határozzák meg az elemzést. (5) Nem a mondatot, hanem az akár több mondatból álló *megnyilvánulást* tekintjük reprezentálandó alapegységnek, lehetővé téve a mondaton belüli és mondatok közötti anaforikus viszonyok egységes kezelését. (6) Ennek megfelelően, illetve a különböző jelenségek egyidejű kezelése miatt a *reprezentáció* nem feltétlenül fa, hanem egy akár különböző típusú éleket tartalmazó összefüggő gráf. Az elvi megalapozást követően az elemző pilot megvalósítását is bemutatjuk. A pilot program az alapelveket szemlélteti és emellett néhány, az ismertetett elveknek megfelelő elemzési lépést is elvégez.

1. Bevezetés

A nyelvészet utolsó évtizedeiben egyeduralkodónak mondható generatív modellek informatikai szempontból nem igazán nyújtanak hatékony megoldást a valóságban előforduló, azaz a nem feltétlenül tökéletesen szerkesztett szövegek elemzésére. Ennek az is az oka, hogy a Chomsky [1] által bevezetett és az ezt követő

generatív technikákban a transzformációk nem invertálhatók, de ez nem lehet a fő ok, hiszen léteznek Chomskyétól eltérő, transzformációmentes generatív modellek is. Ám mindegyik esetében igaz, hogy ezekben a modellekben a „hatékony elemezhetőség" nem a generatív közelítésben preferált kompetencia, hanem a performancia érdeklődési körébe tartozik. A *performanciaalapúság* számunkra azt jelenti, hogy minden nyelvi megnyilatkozás feldolgozandó, ami „előfordul"; viszont ami elvben ugyan lehetne, de valójában nem fordul elő, az valamilyen értelemben kevésbé lényeges. Az emberi nyelvfeldolgozás a nyelvi megnyilatkozással egy időben – ha tetszik: balról jobbra – halad, és igyekszik minden olyan információt felhasználni, mely a megnyilatkozás értelmezéséhez szükséges, még akkor is, ha az – a hagyományos grammatikai értelemben – nem feltétlen tökéletesen szerkesztett. Nincs tehát mód a megnyilatkozások még el nem hangzott, vagy le nem írt részére hivatkozni, azaz legfeljebb feltételezni, valószínűsíteni lehet bizonyos még meg nem jelent összetevőket a már elhangzottak, leírtak alapján, egészen addig, míg a megnyilatkozás be nem fejeződik. Ez nem jelenti azt, hogy nem léteznek olyan megnyilvánulások, amelyek a legvalószínűbbnek tűnő elemzési megoldást „kijátszva", olykor visszalépéses működésre kényszerítik az emberi elemzőt is, ám ezeket úgy tűnik, hogy a hétköznapi kommunikációban a grice-i maximák [2] betartásából következően a kommunikációban kerüljük, és inkább csak viccek, vagy szándékos félrevezetés alkalmával fordulnak elő. Ennek a bizonyítására nagyméretű szövegkorpuszokat kezdtünk építeni[4].

2. Elméleti háttér, összevetés más rendszerekkel

Az elemző architektúrájának kialakításához először megvizsgáltuk a legfontosabb performanciaalapú elemzőket [4], továbbá azokat az elméleti közelítéseket is, melyek a hatékony elemezhetőséget a kompetencia körébe sorolják [5], és azt láttuk, hogy a ma ismert számítógépes mondatelemzők szinte kizárólag egyirányú feldolgozást végeznek, azaz nincs oda-vissza kapcsolat a különböző nyelvi szintek között. Ez a hibák felhalmozódásához vezet, amire általában egy egyszerű gyakoriságon alapuló szűrő a gyakorlati megoldás. A megvalósítandó analitikus grammatikai (a továbbiakban: AnaGramma) elemző viszont párhuzamos szálakon többféle nyelvi elemzést indít, melyekkel egyidejűleg jelennek meg más, a feldolgozandó szöveghez kapcsolható jelentést és világismeretet kezelő szálak. Elemző algoritmusunk tehát egyfajta konszenzust keres a különböző „tudások" között [6]. Amint tehát a humán információfeldolgozásban, a mi elemzőnkben is egyidejűleg és szorosan működnek együtt a nyelvi elemzést és az értelmezést végző modulok (amik a valóságban egy-egy agyi területnek felelnek meg [7]).

Mivel a tervezett reprezentáció legközelebb a függőségi leírásokhoz áll, megvizsgáltuk a hagyományos, kompetenciaalapú világ különböző, létező, hatékony

[4] Szeretnénk a kialakítandó elemző „súlypontját" is a megfelelő helyre tenni, ezért a nagy korpuszok építésére és feldolgozására irányuló kutatásunk egy másik célja a modern grammatikaelméletek által sokat vizsgált, sokszor igen bonyolult – de a hétköznapi életben meglehetősen ritka – nyelvi szerkezetek előfordulási gyakoriságainak vizsgálata. [3]

függőségi elemzőit is, amilyen például a MaltParser [8], a Stanford Parser [9], vagy a véges állapotú függőségi elemző [10]. Ezek valóban a nyelvi egységek egymás közötti viszonyainak leírását célozzák meg, de olyan erősen kötődnek az egymás után következő mondatok szeparált feldolgozásához, hogy nem találtuk őket közvetlenül felhasználhatónak. A magyar nyelvre egyébként történtek korábban függőségi megközelítések, mind szabályalapúak, mint például a holland DLT rendszerhez készített nyelvtan [11], mind adatorientáltak, mint a Szeged Treebank függőségifa-formátumú változata [12]. Ami viszont a magyar nyelvi jelenségek leírását illeti, az eddig készített legátfogóbb magyar mondatelemző, a *MetaMorpho* fordítórendszer magyar nyelvi elemzőjének szabályrendszere is rendelkezésünkre áll [13], bár az nem a függőségi leíráson alapul. Az összes fenti elemző közös tulajdonsága, hogy ezek egyike sem kezeli megfelelően a többértelműségek feloldását, és meglehetősen rossz a hibatűrésük.

Mint Prószéky [14] utal rá, a nyelvi szerkezetek elemzés közbeni kiválasztása közben hozott döntéseink felül tudják bírálni a lexikont. A korábban kialakított nyelvi ismereteket összegző szótárakat és az eddig leírt szintaktikai szerkezeteket adatbázisként használó szabályalapú elemzők és az egyes szerkezetek korábbi gyakoriságára építő valószínűségi elemzők [15] kizárólag csak a „múltbéli" ismeretekre, múltbéli statisztikákra alapozva tudják meghozni döntésüket. Az AnaGramma-elemzésben egy olyan megoldást szándékozunk megvalósítani, melyben az aktuális bemenet esetleges szokatlan felépítését sem a korábbi statisztika támogatásának a hiánya, sem a mechanikusan alkalmazott szabályok sokszor félrevezető elemzési kimenete nem „zavarja meg". Kiinduló hipotézisünk az, hogy a nyelvhasználó fejében két rendszer él: egy a tanult szerkezetekre építő és egy aktuális döntéseket hozó, mely az elhangzó nyelvi elemek valós idejű feldolgozását akkor is képes megvalósítani, ha a „megtanult" szerkezetek egymásnak ellentmondó (például egymáshoz nem illeszkedő jegyszerkezeteket tartalmazó) nyelvtani információkat hordoznak.

A felsorolt eszközök egyike sem kezeli helyesen a többértelmű szerkezeteket és rossz a hibatűrésük, így az újraírószabály-alapú rendszerekben egyetlen nem ismert szó, vagy egy szokatlan, a rendszer számára ismeretlen fordulat az egész elemzés kudarcát okozhatja. A jelenleg kialakítás alatt álló és az elemzésre, mint elsődleges feladatra összpontosító AnaGramma ezzel szemben

1. egyidejűleg több szálon, időben monoton halad (a valódi emberi feldolgozást jobban közelítve), gyakorlatilag visszalépés nélkül (de ennek lehetőségét nem zárja ki)[5];
2. nem tárol „fölöslegesen hosszú ideig" később nem használandó elemzési ágakat (de ez a „hosszú idő" persze szerkezetenként nagyon különböző lehet);
3. mindezekkel együtt, illetve mindezek ellenére: az emberi információfeldolgozáshoz hasonlóan (ha azzal nem is összemérhető mértékben) gyors; és
4. a „hiba" fogalmát nem ismeri, vagyis csak az aktuálisan adott toleranciaszint (ami egy külső paraméter) szerint kezelhető elemei vannak (ezáltal dolgozhatóak föl a helyesírási hibák, az agrammatikus szerkezetek, a szokásos emberi hibák, esetleg beszélt nyelvi átiratok, vagy a nem-anyanyelvűek szövegei is).

[5] ezért egy inkrementálisan balról jobbra haladó elemzőt készítettünk kiindulásként

Elemzőnkben tehát szakítunk a hagyományos „pipeline" architektúrával. A legfőbb ok, hogy a hagyományos architektúrákban az alacsonyabb szinteken képződött hibák javítás nélkül kerülnek át magasabb szintekre és felerősödnek, ezzel rontva a későbbi modulok kimenetének minőségét. Az AnaGrammában több feldolgozási szint (pl. morfológia, igei szerkezetek felismerése, korpuszgyakoriságok, ontológiák és világismeretek) párhuzamosan működnek külön-külön *erőforrásszálként* kiegészítve vagy éppen felülbírálva egymást, minden egyes elemzési lépésben. Az alapelveinkből az is következik, hogy az elemzés folyamán nem használhatunk olyan tradicionális értelemben vett POS-taggert, ami globális információ felhasználásával dönt a mondat minden eleméről. Ehelyett egy olyan n-gram modellt használunk, ami ugyan rendel valószínűségeket az aktuális szóhoz kapcsolható címkékhez, ám csupán az elhangzott, illetve leírt, az aktuális pozíciótól tehát balra álló, azaz a *megelőző* szavak alapján.

Megvizsgáltuk azt is, hogy mely nyelvi elemek indítanak el szövegértelmezés közben valamilyen literális vagy kategoriális predikciót. Néhány ilyen „üzenet" részletes elemzése alapján arra jutottunk, hogy a lehetséges alternatív ágak egyikén-másikán néhány lépés után nem folytatódik az elemzés. Megjegyezzük, hogy bár ez a jelenség a hagyományos táblázatos elemzők [16] világából ismert, azok nem tesznek különbséget a szerkezetek közt aszerint, hogy ezek közül melyik mennyire tipikus, vagy épp mennyire ritka. A tervezett elemző az emberi nyelvfeldolgozás hatékonyságából kiindulva igyekszik elkerülni a kombinatorikus robbanást is, ezért használja az előismeretek összegzéseként kialakított statisztikát: a gyakori szerkezetek sokszor elemzés nélkül, kész belső szerkezettel jelennek meg a feldolgozásban. Informatikai szakszóval ezt gyorsítótárazásnak (angolul cache-elésnek) mondanánk, ám a jelenség a pszicholingvisztikában is jól ismert, és az emberi nyelvértelmezés esetében ezt *egészleges feldolgozásnak* nevezik.

Az eddig megvalósított kompetenciaalapú modellek a nem nyelvi információfeldolgozó alrendszerekkel „természetüknél fogva" semmilyen együttműködést nem feltételeznek. Kijelenthetjük viszont, hogy a performancia nem választható el más kognitív folyamatoknak a nyelvre gyakorolt hatásától, ezért az első elemzési lépéstől kezdve az AnaGramma-módszer a nyelvi, és a modell kidolgozottságától függően bizonyos nyelven kívüli modulok (világismeret, hangulat stb.) párhuzamos kezelésére épít. Ráadásul, a szokásos megoldásoktól eltérően, nem egyes mondatokat, hanem teljes „megnyilvánulásokat" (egy gondolategységet átfogó, általában bekezdésnyi szövegeket) dolgozunk fel, hiszen egy-egy konjunktív elem jelenléte vagy hiánya nem okozhatja az azonos tartalom felszíni különbségek miatti radikálisan különböző feldolgozását, pusztán a mondathatárok különbözősége miatt.

Az egyes mondatok reprezentációi nem a szokásos fastruktúrákban képzelendők el, hiszen a részszerkezetek teljes összekapcsolása nem feltétlen egyetlen mondaton belül valósul meg, továbbá a referenciális elemek is ugyanezen reprezentációban megjelenő, de a hagyományos generatív felfogástól eltérő éleket vezetnek be a leginkább a függőségi leírásra hasonlító AnaGramma-reprezentációkba. A mondatok egyes részeinek referenciális alapon való összekötése (vonatkozó név-

mások, visszautalások kezelése stb.) egy sajátos összefüggő gráfot eredményez[6]. Kimenetként nem pusztán szintaktikai, hanem szemantikai jellegű információkat is szeretnénk megkapni: az elemző célja beazonosítani az összes szereplőt és eseményt, meghatározva a szükséges koreferencia-viszonyokat is. A rendszer végül is létrehoz egy olyan, a mondatot, illetve a bekezdést reprezentáló összefüggő irányított gráfot, amelynek segítségével válaszolni tud majd az olyan kérdésekre, hogy például ki, mit csinált, hol és mikor. Egy ezen az elven a gyakorlatban is működő pilot megoldás jelenleg a következő magyar nyelvi jelenségeket képes kezelni: elváló igekötő, birtokos szerkezet, tagadás, felsorolás (tekintetbe véve, hogy felsorolás tagjai csak valamilyen szempontból egységes elemek lehetnek), értelmező, illetve a vessző írásjel funkciójának meghatározása (azaz, hogy felsorolásra, közbevetésre, vagy értelmezőre utal-e).

3. Az architektúráról

Az elemző balról jobbra halad végig a nyelvi elemeken, amik a mi jelenlegi megvalósításunkban a szavak. Feldolgozza tehát a soron következő szót, tekintetbe véve az összes futó szál által szolgáltatott információt, majd (a) lezár, (b) elindít vagy (c) változatlanul hagy szükséges szálakat. Ha több szabály illeszkedik egy elemre (pl. egyszerre valaminek a birtoka is, és valamilyen esetben is áll), akkor az összes illeszkedő szabályhoz tartozó *strukturális szálaknak* el kell indulniuk. Ezek a lépések együtt határozzák meg, hogy az adott elemnél mi történjen: valamilyen típusú szál induljon, záruljon le, vagy él keletkezzen két elem között a reprezentációban. A szabályokat egyébként a prototípusban még kézzel „gyártottuk", de a későbbiekben elsősorban a statisztikai feldolgozások kimenetén megjelenő minták segítségével hozzuk őket létre, illetve – mint már említettük – felhasználunk rendelkezésre álló nyelvtani adatbázisokat is (ilyen például a MetaMorpho szintaktikai mintáinak egy része).

Alapvetően kétféle, nyelvi elemek által indítható száltípus látszik szükségesnek. A *felkínálás* jellegű szál információt ad az adott elemről (pl. alanyesetű), míg az *igény* jellegű szál keres egy adott tulajdonságú elemet vagy szálat. Például a birtok igényel egy alanyesetű vagy datívuszos alakot, a névutó egy alanyesetű (vagy megfelelő raggal ellátott) alakot, a névelő az esetragos NP-fejet, a tárgyas ige a tárgyat, amire (mindenképpen) szüksége van. Azt állítjuk, hogy a különböző nyelvi aspektusokat figyelő szálak együttműködésének mellékhatása a morfológiai egyértelműsítés és a kombinatorikus robbanások megelőzése, mely utóbbi jelenség a szabályalapú rendszereknél gyakran felmerül a hosszabb mondatok feldolgozása folyamán, akár még morfológiailag egyértelműsített tokenek esetén is.

A tervezett kimenet a feldolgozott szövegből épített szintaktikai-szemantikus relációk hálózata, ami alapján egy lekérdező rendszer meg fog tudni válaszolni

[6] Projektünkben a fentiek szellemében megindult a fent említett (automatikus) korpuszépítés, a főnévi csoportok és mondatvázak mintázatainak (reguláris) szövegkorpuszokban való vizsgálata [3], az új elemző architektúrájának kialakítása, a reprezentációépítés, sőt, az igevonzatok automatikus szemantikus kategorizálása is [17].

olyan kérdéseket, melyek csak implicite vannak benne az eredeti mondatban. Jelenleg a mondatban felismert és azonosított függőségi relációkból épült egységek fája készül el, kiegészülve azokkal a szemantikai jellegű információkkal, amelyeket az elemzés során a szövegből nyertünk.

Reprezentációnkban előnyben részesítjük a nem fa formájú, hanem általában DAG-formájú függőségi gráfokat. Mivel a *koreferenciák* nem feltétlenül azonos mondatban jelennek meg, így az általuk bevezetett élek „színe" más, így nem tudják elrontani a szerkezetet, azaz miattuk nem kaphatunk irányított, körmentes gráfot eredményül.

Az *ellipszis* jelenségek kezelése miatt megengedjük a szálaknak, hogy túlléphessenek a mondathatáron. Úgy véljük, hogy az elemzésnek nem szabad megállnia a mondatok végén, mert az egymagukban álló mondatokkal szemben a hosszabb megnyilatkozások az emberi kommunikáció természetes egységei. Az egymást követő mondatok témája sokszor azonos, ezért a természetes emberi kommunikáció során lehetséges – és többnyire meg is történik – az egyes elemek kihagyása (az ellipszis jelenség), ami a legtöbb hagyományos elemzőnél komoly problémákat okoz. A rendszerünk által feldolgozandónak szánt nyelvi egységek néhány mondatból álló összefüggő szövegek, ám a sok mondatból álló, nagyobb művek feldolgozását egyelőre nem szándékozzuk megcélozni.

Nagyon fontos számunkra a szövegben előforduló események szereplőinek azonosítása, és a *koreferenciaviszonyok* meghatározása, más szóval annak meghatározása, hogy mely szereplők azonosak a világban („ki kicsoda?"). Más szóval, szeretnénk helyesen kezelni, hogy mely szereplő „új" a szöveg egy adott pontján való megjelenésekor, és mely nyelvtani elem utal egy korábban már megjelent szereplőre, illetve van-e, és ha igen, milyen kapcsolata a korábbiakkal. Lényegében szereplőnek tekinthető az összes névszó, az igeragokból kikövetkeztethető alanyi, tárgyi szereplők, sőt, Davidson nyomán a *neo-davidsoniánus eseményszemantika* [18] elveinek megfelelően maguk az események (azaz az igék) is, mivel vissza tudunk utalni rájuk. Az azonosítás érdekében a korábbi mondatokat és minden korábbi szereplőt folyamatosan nyilvántartunk.

További szálak hasznosítják a *lexikai egységek* és *lexiko-szintaktikus szerkezetek* gazdag leírását, amit a *MetaMorpho* elemző adatbázisait felhasználva építettünk fel. Például egy „felkínálás" típusú szál alapvető szintakto-szemantikus jellemzőikkel (élő, ember, absztrakt stb.) annotálja az egyes egységeket a rendelkezésre álló, mintegy 118 000 szót és többszavas kifejezést tartalmazó adatbázisból. Egy „igény" szál pedig a *MetaMorpho* 35 000 darabos nyílt konstrukciós szabályhalmazából kapcsolatokat javasol az igék, főnevek és melléknevek a lehetséges argumentumai között (például: *eszik valamit, ellenségesség valamivel szemben, érdeklődés valamivel kapcsolatban*). Ezeken túl kísérletezünk még az igei szerkezetben megjelenő argumentumok predikciójával, mely a korpuszbeli adatok (együttes előfordulás, gyakoriság), az ontológiai információk [19] és a lexikon (a *MetaMorpho* elemző igei szerkezetek adatbázisa) információira támaszkodik. Építünk még a *Mazsola* projektből [20] származó ige-főnév együttes előfordulások adatbázisára, és azon is dolgozunk, hogy össze tudjuk kapcsolni

őket a *Magyar WordNettel*, hogy általánosított szemantikai osztályokat találjunk az igei szerkezetek szemantikus szelekciós megszorításai között [17].

4. A rendszer működésének alapelveiről

Új elvű nyelvi elemzőnk kialakításának első lépéseként azonosítottuk a feldolgozás során használni kívánt formális utasítástípusokat. Meg kellett találnunk azokat az elemeket, amik meghatározzák, hogy milyen fajta elemek jöhetnek utánuk. Például a névelőt követő főnévi csoport végén valahol egy főnévnek, egészen pontosan valamilyen főnévi szerepű elemnek kell állnia. Előbb-utóbb megjelennek a szövegben olyan elemek, melyek „kielégítenek" egy korábbi „igényt". Például az igei argumentumok kitöltenek egy helyet a már korábban látott ige vonzatkeretében. Ha az ige maga valamely argumentuma után jön, akkor a korábban megjelent argumentumok – ha megfelelő jegyeik kompatibilisek – automatikusan kitöltik a szerkezetet a megfelelő módon.

Vannak azonban a fentiektől eltérő, más típusú műveletek is: a konjunkciós szerkezetek például csak akkor azonosíthatóak, ha egy konjunktív elem ténylegesen feltűnik. Ez lehet „és", „vagy" vagy épp egy erre szolgáló vessző, mert ezek vezetik be a konjunktív szerkezet következő tagját. Ha a rendszer felismer egy ilyen elemet (de csak akkor!), módosítania kell az utolsóként feldolgozott elem reprezentációját a felismert szerkezetnek megfelelően, hiszen az előző elem volt ennek a konjunktív szerkezetnek az első tagja, amit az előző lépésben, annak feldolgozásakor még nem tudhattunk róla. A konjunkciót egyébként egyetlen egységként kezeljük, anélkül, hogy állást foglalnánk arról, hogy van-e az ilyen szerkezeteknek feje. Jelenleg épp az ilyen, exocentrikus szerkezetekre vonatkozó műveletek balról jobbra történő feldolgozásának formalizálásán dolgozunk.

Pilot implementációnkban megpróbálunk kezelni néhány olyan gyakori, alapvető jelenséget, amiket nem feltétlenül egyszerű kezelni más keretrendszerekben. Ilyenek például

- az elváló igekötő és az igető, illetve a birtokos szerkezetek részeinek összekapcsolása,
- a felsorolások/koordinációk (amik azonos típusú elemekből állnak) komplex egységként való felismerése,
- a vessző szerepének felismerése aszerint, hogy mit vált ki: mellékmondatot, felsorolást, zárójeles kifejezést/közbevetést vagy értelmezőt,
- a tagadás hatókörének felismerése.

Elemzőnk elkészült prototípusát *újsághírek összefoglalóin* teszteltük, melyeket a `www.inforadio.hu` RSS csatornájáról töltöttünk le. A két-három mondat hosszúságú hírek általában egyetlen politikai vagy gazdasági eseményt írnak le. Nyelvi komplexitásuk közel áll ahhoz, amit modellezni szeretnénk, ezért megfelelő bemenetül szolgálnak az elemző számára. Először a bemeneti szöveget előfeldolgozásként lemmatizáltuk (ezzel mintegy modelláltuk a flektáló nyelvek toldalékolt alakjainak „szótári lookup" jellegű kezelését). A morfológiai többértelműségek ezen a szinten természetesen meg kell, hogy maradjanak, mert – mint

korábban említettük – nem használhatjuk a jól ismert egyértelműsítő eljárásokat, mivel azok általánosságban megsértik a monoton balról jobra haladó elemzést. Néhány rövidhír részletes elemzése alapján arra jutottunk, hogy egyfajta dinamikus (azaz az aktuális szó kategóriájától függő) előrenéző stratégiát érdemes használnunk, ugyanis a legtöbb alternatív elemzési ág gyorsan befejeződik, mert a különféle szálak nem engedik folytatódni őket néhány lépés után. Megjegyezzük, hogy ez a jelenség jól ismert a hagyományos táblázatos elemzőknél, de azok nem képesek különbséget tenni a különböző struktúrák között azok tapasztalati gyakorisága alapján. Mi ezeknél a döntéseknél állandóan tekintetbe veszünk egy olyan korpuszgyakorisági szálat, mely a háttérben fut és egy nagy korpusz adataira támaszkodik. Ez tájékoztat arról, hogy a meglévő elemzés mennyire felel meg a szokásos mintázatoknak, illetve döntési helyzetben segít választani több lehetséges alternatíva közül. Mindig csak a (balról jobbra) soron következő szót értékeljük ki, figyelve, hogy milyen gyakorisági viszonyban áll az eddigiekkel. Például az *esik* alak után alanyként a *szó* meglehetősen jól elfogadható, mert kb. 15%-ot képvisel az *esik* mellett. Ha viszont ezen a lépésen is túl vagyunk, akkor már nagyon várjuk a *-rÓl* ragos alakot, mivel az 90%-os valószínűségű az *esik szó* kifejezés esetében.

Fontos, hogy a rendszer kategóriáinak kialakításánál csak a szükséges általánosításokat tegyük meg. Például adott esetben létrehozhatunk olyan – a hagyományos nyelvtani kategóriáktól eltérő – szófajt, amely adott esetben egyetlen kivételes szót (pl. *is*) tartalmaz, vagy dönthetünk úgy, hogy az alany- és a birtokosesetet a többi esettől teljesen elkülönítve, új néven kezeljük.

5. Összefoglalás

Kutatásunk egy pszicholingvisztikai motivációjú, performaciaalapú, párhuzamos feldolgozást végző nyelvi elemzőt céloz meg. Megpróbáltuk összegyűjteni az ehhez a működéshez szükséges ismereteket az irodalomból, de azt találtuk, hogy az emberi nyelvfeldolgozás általunk vizsgált aspektusait egyetlen ma működő elemző sem elégíti ki megfelelően, így lefektettük egy új elképzelés, az AnaGramma alapjait. A kidolgozott elvek működtetéséhez első lépésként egy minimális képességű, de a működés alapjait mégis bemutatni képes pilot programot is készítettünk, melynek forráskódja megtalálható az alább internetes oldalon:
`https://github.com/ppke-nlpg`.

Köszönetnyilvánítás

Köszönjük a TÁMOP-4.2.1.B – 11/2/KMR-2011–0002 és a TÁMOP: 4.2.2/B – 10/1–2010–0014 projektek részleges támogatását.

Hivatkozások

1. Chomsky, N.: Syntactic structures. The Hague:Mouton (1957)
2. Grice, H.P., Harman, G.: Logic and conversation. Encino:Dickenson (1975)
3. Endrédy, I., Novák, A.: Egy hatékonyabb webes sablonszűrő algoritmus – avagy miként lehet a cumisüveg potenciális veszélyforrás Obamára nézve. A IX. Magyar Számítógépes Nyelvészeti Konferencia előadásai (2013) 297–301
4. Bunt, H., Merlo, P., Nivre, J.: Trends in Parsing Technology. Dordrecht:Springer (2010)
5. Pritchett, B.L.: Grammatical competence and parsing performance. University of Chicago Press (1992)
6. Pléh, Cs.: Mondatmegértés a magyar nyelvben. Osiris Kiadó, Budapest (1999)
7. Csépe, V.: Az olvasó agy. Akadémiai Kiadó, Budapest (2006)
8. Nivre, J.: Inductive dependency parsing. Springer (2006)
9. De Marneffe, M.C., MacCartney, B., Manning, C.D.: Generating typed dependency parses from phrase structure parses. In: Proceedings of LREC. Volume 6. (2006) 449–454
10. Oflazer, K.: Dependency parsing with an extended finite-state approach. Computational Linguistics **29**(4) (2003) 515–544
11. Prószéky, G., Koutny, I., Wacha, B.: A dependency syntax of Hungarian. Metataxis in Practice (Dependency Syntax for Multilingual Machine Translation) (1989) 151–181
12. Vincze, V., Szauter, D., Almási, A., Móra, Gy., Alexin, Z., Csirik, J.: Hungarian dependency treebank. In: LREC. (2010) 1855–1862
13. Prószéky, G., Tihanyi, L., Ugray, G.: Moose: A robust high-performance parser and generator. Proceedings of the 9th Workshop of the European Association for Machine Translation (2004) 138–142
14. Prószéky, G.: Számítógépes morfológia. In Kiefer, F., Bánréti, Z., eds.: Morfológia (Strukturális magyar nyelvtan III). Volume 3. Akadémiai Kiadó, Budapest (2000) 151–1064
15. Brants, T., Crocker, M.: Probabilistic parsing and psychological plausibility. In: Proceedings of the 18th conference on Computational linguistics-Volume 1, Saarbrücken:Association for Computational Linguistics (2000) 111–117
16. Révész, G.: Bevezetés a formális nyelvek elméletébe. Akadémiai Kiadó, Budapest (1979)
17. Miháltz, M., Sass, B., Indig, B.: What do we drink? Automatically extending Hungarian WordNet with selectional preference relations. In: Joint Symposium on Semantic Processing. (2013) 105–109
18. Terence, P.: Events in the semantics of English: A study in subatomic semantics (1990)
19. Miháltz, M., Hatvani, C., Kuti, J., Szarvas, Gy., Csirik, J., Prószéky, G., Váradi, T.: Methods and results of the Hungarian WordNet project. In Tanács, A., Csendes, D., Vincze, V., Fellbaum, C., Vossen, P., eds.: Proceedings of the Fourth Global WordNet Conference (GWC-2008), Szeged, University of Szeged (2008) 311–321
20. Sass, B.: The Verb Argument Browser. 11th International Conference on Text, Speech and Dialog (TSD) (2008) 187–192