Pécsi Tudományegyetem Bölcsészettudományi Kar

Nyelvtudományi Doktori Iskola

Alkalmazott Nyelvészet Program

A TOTÁLIS LEXIKALIZMUS ELMÉLETÉTŐL A KÍSÉRLETI IMPLEMENTÁCIÓIG

PhD értekezés

Tézisek

Kleiber Judit

Témavezető: Dr. Alberti Gábor

Pécs, 2008.

A tézisfüzet felépítése:

I. A kutatási feladat rövid összefoglalása (cél, eszköz, eredmény)
II. A dolgozat kivonata
III. Tézisek
IV. Hivatkozások
V. A témához kapcsolódó saját publikációk
VI. Függelék: Saját hozzájárulásom a közös munkához

I. A kutatási feladat rövid összefoglalása (cél, eszköz, eredmény)

A dolgozatban arra vállalkoztam, hogy bemutassak egy több évig tartó kutatást, a célokat, eredményeket, tapasztalatokat. A kutatási és programozási munkán túl elvállaltam tehát a „krónikás" szerepét is: a dolgozatban a célom alapvetően annak megmutatása volt, hogy a kutatás miért volt hasznos, miért érdemes a totálisan lexikalista megközelítést elméletben és gyakorlatban is alkalmazni. A munkát mindig egy csapat végezte, ezért nehéz egy konkrét személyhez kötni egy adott eredményt. Erre a dolgozatban nem is vállalkoztam, viszont a tézisfüzet végén részletesen ismertetem, hogy én mit tettem hozzá a közös munkához.

Cél:
A kutatás célja kettős:

a) Elméleti: a totálisan lexikalista GASG (Generatív Argumentumstruktúra Grammatika) létjogosultságának és egzaktságának bizonyítása.
b) Gyakorlati: olyan elemző és gépi fordító megalkotása, amely minden eddiginél alaposabb, köszönhetően a nagyon erős szemantikai komponensének.

Eszköz:
A célok megvalósításának eszköze: az elmélet *implementációjának* elkészítése.
GeLexi-projekt: Prolog programnyelven, alapvetően az elméleti cél megvalósítása
LiLe-projekt: SQL adatbázis és Delphi program, hatékonysági szempontok is

Eredmény:
A GeLexi-projekt az első célt megvalósította, bebizonyította, hogy kis adatbázison a mechanizmusok működnek, a program elemez és fordít. A jól formált (angol vagy magyar nyelvű) mondatokhoz morfofonológiai, szintaktikai és szemantikai reprezentációt társít, és a két nyelv között a gépi fordítást is megvalósítja. Erről szól a dolgozat legnagyobb része.

Következő lépésként a LiLe-projekt új (relációs) adatbázist épített, mellyel a megközelítés hatékonyságát kívánta vizsgálni (inkább a gyakorlati cél). A megvalósítás a morfofonológiai komponensig jutott, amikor is a két projekt egyesülésével létrejött a ReALIS-projekt, amely jelenleg a szintaktikai és szemantikai elemzés és reprezentáció megvalósításán dolgozik a LiLe-projekt által kidolgozott adatbázis-szerkezetet is felhasználva.

II. A dolgozat kivonata

1 Bevezetés

1.1 A dolgozat tárgya

Dolgozatomban egy négyéves számítógépes nyelvészeti kutatás tanulságairól számolok be. Célunk az volt, hogy kipróbáljunk egy új típusú grammatikát elméletben és gyakorlatban egyaránt. Nyelvtanunk megvalósítja a „totális" lexikalizmust, ami a végsőkig fokozása az elmúlt évtizedekben megfigyelhető hangsúlyeltolódásnak szintaxis és lexikon között. Minden információt a szótári komponens tartalmaz, így nincs szükség frázisstruktúra-szabályokra. Az egyetlen művelet az *unifikáció*, amely a mondatok összeépítésének a motorja.

Az elmúlt években a nyelvtan implementációján dolgoztunk, készült egy elemző Prolog programnyelven, amely kis adatbázison magyar és angol mondatokat elemez, hozzájuk szemantikai reprezentációt társít, illetve köztük a gépi fordítást is megvalósítja (GeLexi-projekt). Miután úgy tűnt, a mechanizmusok működnek, hatékonyabb adattároláson kezdtünk dolgozni (SQL adatbázis), hogy lássuk, hogyan birkózik meg a totális lexikalizmus nagyobb mennyiségű lexikai egységgel (LiLe-projekt). Az eredményeink bíztatóak, de adatbázisunk mérete még nem érte el azt a szintet, hogy az ismert módszerekkel értékelni lehessen, ezért az újabb négyéves kutatás célja tovább haladni ezen az úton, és kipróbálni a totálisan lexikalista megközelítést olyan méretű adatbázison, amely már megmutathatja mechanizmusunk hatékonyságát is (ReALIS-projekt).

1.2 Célok

Elméleti célunk annak igazolása, hogy a kiinduló nyelvtan (a GASG, Alberti 1999) egy jól formalizált, egzakt rendszer; illetve hogy a totálisan lexikalista megközelítés hasznos eszköze lehet a nyelvleírásnak. Az elmúlt évtizedekben a korábbi szintaxisközpontú grammatikák irányából erőteljesen a szótári komponens minél részletesebb kidolgozása felé tolódott el a hangsúly. A lexikalista elméletek sikeressége azt mutatja, hogy érdemes ebben az irányban vizsgálódni, és kipróbálni egy totálisan lexikalista grammatikát.

Gyakorlati célunk létrehozni egy nyelvelemző programot a GASG alapján, amelynek a központi komponense egy (diskurzus)szemantikai reprezentáció, így alkalmas olyan magasabb szintű nyelvtechnológiai célokra is, mint például a kérdés-megválaszolás, vagy a jó minőségű gépi fordítás. Széles körben elfogadott nézet, hogy intelligens alkalmazásokhoz nem elégségesek a pusztán statisztikai alapokon működő rendszerek, azokhoz már elméleti (nyelvészeti) alapokon nyugvó alkalmazásokra van szükség. Programunktól azt várjuk, hogy teljesítse a generatív alapfeladatot: a beírt szósorról tudja eldönteni, hogy grammatikus-e, és amennyiben az, rendeljen hozzá kimenetként morfológiai, szintaktikai és – ami a legfontosabb – diskurzus-szemantikai reprezentációt. Mivel a mechanizmusok nyelvfüggetlenek, rendszerünktől azt várjuk, hogy bármely nyelv lexikai egységein működni tud, és bármely két nyelv között képes a fordításra (nem kell minden nyelvhez külön mechanizmust írni az elemzéshez, illetve minden nyelvpárhoz külön algoritmusokat definiálni a gépi fordításhoz).

Projektünkben tehát elméleti és alkalmazott nyelvészeti célok fonódtak össze elválaszthatatlanul. A grammatikát a számítógépes implementálhatóság legitimálja (célunk ezt megmutatni), másrészt minden nyelvészeti ötlet azt a célt is szolgálja, hogy minél jobb, minél többféle célra felhasználható nyelvelemző rendszerünk legyen. Munkánkkal azt is szeretnénk bizonyítani, hogy a számítógépes nyelvészetnek érdemes visszafordulnia a tiszta (generatív) nyelvelméleti alapok felé.

1.3 Eredmények

Jelenlegi elemzőnk néhányszáz szavas adatbázison működik. A lexikai egységek morfémák, vagyis tövek és toldalékok, összes (fonológiai, morfológia, szintaktikai és szemantikai) tulajdonságukkal együtt. A program a beírt mondatokhoz négyféle reprezentációt társít: morfológiai (szavak morfémákra bontása), szintaktikai (régens-vonzat viszonyok, szabad bővítmények), szemantikai (egy DRS); továbbá a mondat ún. kopredikációs hálózatát is elkészíti, amely egy szintaxis és szemantika közötti szint, és a gépi fordítást segíti.

A program magyar és angol nyelvű mondatokat elemez, és köztük a gépi fordítást is megvalósítja, mindkét irányban. Ehhez alapvetően a kopredikációs hálózatot használja, amely nagyon hasonló az azonos jelentésű magyar és angol mondatok esetében. Ha kérjük (és van), több megoldást is ad.

Az elemzőt Prolog programnyelven írtuk. A Prolog egy magas szintű programnyelv, sok számítógépes nyelvészeti projekt használja. Mi alapvetően azért választottuk, mert – akárcsak a GASG esetében – működésének motorja az unifikáció.

A totális lexikalizmus elvének gyakorlatban való kipróbálásán két projekt is dolgozott. Az első programot a GeLexi-pojekt készítette, a fő cél az elmélet igazolása volt. Később a LiLe-projekt új (SQL-)adatbázist épített a technológiai szempontok figyelembe vételével (bővíthetőség, rugalmasság, webes megjeleníthetőség), amelyhez Delphi programnyelvű elemző tartozott. A lexikon részletes kidolgozása csak a morfológiai szintig jutott, és csupán néhányszáz lexikai egységet rögzítettünk. Ennek oka, hogy 2006-ban a két projekt egybefonódott, és jelenleg új lexikon és program készül az eddigi tapasztalatok alapján (ReALIS-projekt). A cél továbbra is kettős: az elmélet igazolása és egy hatékony nyelvelemző (és gépi fordító) program megalkotása.

2 Nyelvtechnológia

2.1 Elvárások

A számítógépes nyelvészet kialakulása az ’50-es évekre tehető, amikor a gépi fordítás igénye először felmerült, és az akkori kutatók számára megvalósíthatónak tűnt. Hamarosan be kellett látniuk azonban, hogy a probléma sokkal bonyolultabb, mint először képzelték, sok kutatást és rengeteg részfeladat megoldását igényli. Ekkor kezdtek mind a számítógépes szakemberek, mind a nyelvészek egyre nagyobb számban ezzel a területtel foglalkozni. Az elmúlt évtizedek alatt számos problémára születtek is megoldások, működő programok, de sok közülük máig is megoldatlan, mint például a gépi fordítás maga. Léteznek ugyan gépi fordító programok, de hatékonyságuk távolról sem éri el a kívánt szintet.

A kitűzött célok eléréséhez két oldalról is szükség volt fejlődésre: a nyelvészeti elméletek és a technológia oldaláról. Elméleti oldalról konzisztens, formalizálható, így könnyen és hatékonyan implementálható nyelvészeti elméletekre, rendszerekre volt szükség (GPSG, LFG, véges állapotú eszközök fonológiára, morfológiára); illetve arra, hogy a nyelvi szintekről egzakt, formális leírások szülessenek: a cél nem csupán a *fonológia*, a *morfológia* és a *szintaxis* kidolgozása, hiszen egy igazán intelligens programnak *szemantikai*, sőt *pragmatikai* információt is tartalmaznia kell. Számos elmélet törekszik valamiféle *univerzalitásra*, vagyis hogy a különböző nyelveket egységes keretben lehessen leírni, így a kapott elemzések minél nagyobb mértékben párhuzamosak legyenek egymással, ami a többnyelvű alkalmazások (például gépi fordítás) fejlesztését jelentősen megkönnyítik, a minőségét és a hatékonyságát pedig nagymértékben javíthatják.

A technológia oldaláról pedig olyan praktikus módszerek kellettek, amelyekkel hatékonyabbá tehetők a számítógépes programok (leginkább különféle statisztikai módszerek, újabban tanító algoritmusok, korpuszok használata). Ki kellett továbbá dolgozni a

részfeladatok körét, amelyekre bontva könnyebben elérhetőnek tűnt a kiinduló cél – a gépi fordítás – megvalósítása. A legtöbb számítógépes nyelvészeti projekt egyszerre egy jelenségre koncentrál, egy probléma megoldására fejleszt minél pontosabb és hatékonyabb programot. A megoldandó részfeladatok között szerepel például a *szegmentálás* (szövegek szavakra, szavak morfémákra bontása), *kategorizálás* (pl. szavak vs. számok, szavak szófajokba sorolása, tulajdonnevek felismerése), *elemzés* (morfológiai: tő és toldalékok, szintaktikai: ige és vonzatai), különféle *egyértelműsítések* (szószintű, mondatszintű), *anaforák* referenciájának megtalálása (pl. egy névmás mire/kire vonatkozik), *beszédfelismerés*, *beszéd-előállítás*, természetes nyelvi *szöveg előállítása* stb.

Egy nyelvelemző rendszertől fontos elvárás, hogy a generatív alapfeladatot teljesíteni tudja, vagyis a beírt mondatról el tudja dönteni, hogy grammatikus-e (és különféle reprezentációkat társítson hozzá). Azonban nem minden rendszer tűzi ezt ki célul, hiszen legtöbb esetben a felhasználó nem arra kíváncsi, hogy jól formált-e az adott mondat, hanem arra, hogy milyen információt közvetít. Egyre több hibás szöveggel találkozhatunk (például az interneten), ezért fontos, hogy – ha végez is az adott rendszer grammatikalitás-ellenőrzést –, azokhoz a mondatokhoz is tudjon reprezentációkat (esetleg fordítást) társítani, amelyek nem felelnek meg minden szempontból az adott nyelv szabályainak (Prószéky 2005).

A gépi fordítás mellett időközben természetesen egyéb célok is megfogalmazódtak, amelyek elérését az imént említett folyamatok segítik. Ilyen például a gépi fordításhoz kapcsolódó egyéb szoftverek létrehozása, amelyek (főleg az interneten fellelhető) idegen nyelvű szövegek megértését segítik, mint például a többnyelvű elektronikus szótárak. De fontos cél olyan programok létrehozása is, amelyek képesek *visszakeresni* vagy *kinyerni információt* szövegekből, *összefoglalni* szövegrészek tartalmát, *kérdés-válasz* dialógusokat kezelni, így (akár írásban, akár szóban működő) *interaktív rendszerek* alapját képezni, vagy *multimédiás eszközök* megalkotását és használatát sokoldalúbbá és kényelmesebbé tenni. Fontos cél továbbá az oktatás támogatása, olyan szoftverek létrehozása, amelyek például a *számítógéppel segített nyelvtanulást* (vagy bármi más tanulását) hatékonyabbá tudják tenni.

2.2 Módszerek

A programok hatékonyabbá tételét segítő módszerek közül a legrégebben és legáltalánosabban a különféle *statisztikai módszereket* használják. Szinte minden nyelvi szinten alkalmazhatók, és jelentősen gyorsabbá teszik a szöveg feldolgozását. Egy másik terület, amely az utóbbi években egyre nagyobb teret hódít, a *korpusznyelvészet*. A mondatok elemzésekor nagyon hasznosnak bizonyult óriási méretű (több millió szavas) korpuszokat használni, amelyeket különböző mértékben annotálnak (leginkább kézzel), vagyis a lexikai egységekhez különféle morfológiai, vagy akár szintaktikai, sőt szemantikai információt társítanak. Minél alaposabban annotált egy korpusz, annál jobban használható, viszont annál nagyobb munka előállítani. Intelligensebb programokhoz különféle *ontológiák* használata is szükséges lehet, amelyek a világról való tudásunkat tárolják valamilyen formában. Végül az utóbbi években terjedt el szélesebb körben a különféle *tanuló algoritmusok* használata, hiszen a leghatékonyabb az lenne, ha a számítógép is – akárcsak az ember – valamennyi inputból tanuló képessége segítségével meg tudná tanulni a nyelvet.

Két fő megközelítés létezik az általunk is kitűzött célokra (komplex nyelvi elemzés, fordítás): a *szabály alapú* és a *korpusz alapú*, amelyek közül a számítógépes nyelvészet korábbi szakaszára az előbbi, míg az elmúlt néhány évre inkább az utóbbi megközelítés volt jellemző. Mindkettőnek vannak hiányosságai, emiatt napjainkban az igazán hatékony rendszerek mindkettőt alkalmazzák. A szabály alapú rendszerek esetében az adatrögzítés lassú, rengeteg munkaórát igényel; illetve bizonyos jelenségek (például a többértelműség) kezelését nem is igazán lehet szabállyal megragadni, hatékonyabb példákat sorakoztatni, vagy a kontextusra támaszkodni. A korpusz alapú rendszerek esetében lényegesen rövidebb idő

alatt lehet nagy mennyiségű adathoz jutni, azonban a megközelítés akkor a legeredményesebb, ha a felhasznált korpuszok alaposan annotáltak (mint például a treebankok, amelyek nem csupán morfológiai, hanem szintaktikai vagy akár szemantikai információt is tartalmaznak), ami viszont szintén sok befektetett munkát igényel, ezért számos nyelvre nem is léteznek ilyen adatbázisok. További hátránya a korpusz alapú megközelítésnek, hogy (akármekkora is a méretük) nem minden lehetséges nyelvi forma található meg bennük; illetve hogy intelligensebb célokra nem igazán alkalmasak. A két eszközt kombináló hibrid rendszerek igyekeznek mindkét megközelítés előnyeit kihasználni, így a korábbiaknál pontosabb alkalmazásokat tudnak viszonylag rövid idő alatt fejleszteni.

Különbség van a szabály alapú rendszerek között abban, hogy mennyire alaposan elemzik a mondatokat (shallow vs. deep parsing). A *sekélyelemzés* előnye, hogy gyors és robusztus, hátránya viszont, hogy mivel csupán részleges elemzést végez, nem annyira precíz (Frank 2003), így komplexebb célokra (mint például az igazán jó minőségű gépi fordítás) nem alkalmas. A *mélyelemzést* végző rendszerek sokkal alaposabbak és pontosabbak, és az utóbbi időben lefedettségben is felveszik a versenyt a sekélyelemző rendszerekkel.

Egyetértés mutatkozik abban, hogy ha intelligens alkalmazások megalkotása a cél (például szövegek lényegének összefoglalása, jó minőségű gépi fordítás), akkor arra a szabály alapú, mélyelemzést végző rendszerek a legalkalmasabbak, mert pontosabbak, és mert képesek szemantikai reprezentációt társítani a mondatokhoz. A kezdeti nagyobb energia-befektetés (nyelvészeti alapok, kézi adatrögzítés) pedig megtérül később, például amikor új nyelvekre dolgozzák ki a rendszert (Forst et al. 2005). További előnyük az újra-hasznosíthatóság, vagyis hogy az ilyen módszerrel készült rendszerek más alkalmazásokban is használhatók (például egy elemző gépi fordításra). A megfelelő hatékonyság elérése érdekében azonban statisztikai módszerek bevonására is szükség van, és (minimálisan adattöltésre és egyértelműsítésre) korpuszok használata is elengedhetetlennek látszik.

3 Lexikalizmus

3.1 Elméletben

A generatív nyelvészet kezdeti szakaszában a mondatok grammatikalitásának vizsgálata során a szintaktikai szerkezetnek jutott a vezető szerep, míg a szótári komponens feladata pusztán a szavak felsorolásából állt. Ez a megközelítés többé-kevésbé sikeresnek bizonyult az angolhoz hasonló konfigurációs nyelvek esetében (bár a transzformációk létjogosultságát nem sikerült igazolni), a szórend helyett inkább morfológiai eszközöket használó nyelvek esetében azonban nehézkesnek bizonyult.

A hetvenes évektől kezdődően egyre több olyan elmélet született, amely a lexikont helyezi előtérbe: a nyelvi jellegzetességeket inkább a szótárban tárolja, míg a szintaktikai komponens csupán néhány nagyon általános frázisstruktúra-építő szabályt tartalmaz. Ezek az ún. *lexikalista nyelvtanok* eredményesebben kezelik a magyarhoz hasonló szabadabb szórendű nyelveket, miközben a konfigurációs nyelvek leírására ugyanúgy alkalmasak, mint a transzformációs elméletek.

A lexikalista nyelvtanok legfontosabb jellemzője, hogy megszorítás alapúak, vagyis nem egy sikeres deriváció, hanem egy kielégíthető követelményhalmaz a grammatikus nyelvi formák ismérve. A legfontosabb megszorítás alapú elméletek az LFG, a HPSG, a kategoriális nyelvtanok és a konstrukciós nyelvtan. A lexikon irányába való erőteljes elmozdulást jól mutatja, hogy egyes – korábban a lexikonra kis hangsúlyt fektető – elméleteknek, mint például a TAG-nek (Tree Adjoining Grammar) elkészült a lexikalizált változata (LTAG, Lexicalized Tree Adjoining Grammar). A nyelvtan „atyja”, Aravind Joshi, napjainkban már szintén azt vallja, hogy „Grammar ≈ Lexicon” (Joshi 2003).

A lexikalista elméletek négy legfontosabb jellemzője Trón (2001) alapján a következő:

(1) *Deklarativitás* (megszorítás alapúság), vagyis nem átalakító jellegű szabályokkal, hanem jólformáltsági megszorításokkal dolgozik. Nem a levezetés (deriváció) szabja meg, mi lesz grammatikus; jól formált kifejezés az, ami minden megszorítást kielégít. Kérdés, hogy milyen jellegű megszorításokat enged meg az adott nyelvtan. Annak ellenőrzésére, hogy a különféle jegyekkel bíró elemek kombinálódhatnak-e az adott mondatban, az *unifikáció* mechanizmusát használja.

(2) *Egyszintűség*, vagyis elfogadja, hogy különböző nyelvi reprezentációs szintek lehetnek, de azt vallja, hogy egy jel jólformáltságát ezek a (különböző szinteken lévő) megszorítások egyszerre szabják meg, egyszerre lépnek életbe, vagyis nem lehetnek köztes, rosszul formált reprezentációk (monoton építkezés), és egyik megszorítás sem lehet erősebb, mint a másik (Kálmán et al. 2002). Az egyszintűség mellett több érv is felhozható. Például hogy a pszicholingvisztikai kutatások is inkább ezt támasztják alá, vagy hogy így egységes formalizmus használható az ábrázolásban (Trón 2001). Kálmán et al. (2003) szerint pedig a legnagyobb probléma a hagyományos generatív modellekkel pontosan a modularitásuk: nehéz például jelentéstani információ nélkül igazán alapos és megbízható szintaktikai elemzést végezni, vagy ha a jelentéstant szétválasztjuk a pragmatikától, akkor nagyon bonyolult leírást kellene alkalmaznunk, stb. Olyan szintű együttműködést kellene elvárnunk a moduloktól, hogy az már nem is lenne modularitás, ezért javasol inkább (implementációs célokra is) egyszintű grammatikát.

(3) *Lexikai integritás* elvének betartása, vagyis hogy a szavakat egy független lexikai modul állítja elő, belső szerkezetük a szintaxis számára nem hozzáférhető. Ezt egyes elméletek nem tűzik ki célul, azok, ahol bármely morféma (egy toldalék is) önálló követelményekkel léphet fel. Ilyen elméletet ír le például Gambäck (2005), de ilyen a GASG morféma alapú verziója is. Ezeknek a megközelítéseknek az előnye, hogy számukra nem okoz problémát, hogy különböző nyelvekben különböző helyen található a szószint, így a nyelvi jelenségek egységesebb kezelését biztosítják[1].

(4) *Hierarchikus lexikon* alkalmazása, vagyis hogy a szótár elemei típushierarchiába (öröklődési hierarchiába) rendeződnek, amely a tudásreprezentációk egyik fajtája. „Fent" találhatók az általános elemek, „lent" pedig az egyediek. A csomópontok (fogalmak) leírása általában attribútum–érték párokkal történik. Az általánosabb fogalom jegyei öröklődnek a speciálisabbakra, de azok értékei eltérhetnek, felülírhatják azokat (kivételek). A lexikon tehát strukturált, nem csupán idioszinkráziák tára. Ez a tulajdonság sem igaz minden lexikalista elméletre, nem teljesül például a kategoriális és a konstrukciós nyelvtan (és elméletileg a GASG) esetében sem.

Az egyik legismertebb lexikalista (deklaratív) elmélet az LFG (Lexikai Funkcionális Grammatika), amely nem nyelvészeti értelemben vett szintaxiselmélet, hanem olyan formális keret, amelyen belül több különböző grammatika megfogalmazható (Komlósy 2001). Az univerzális grammatika modellje. Azt vallja, hogy a nyelvi jelek különböző szinteken írhatók le, és a nyelvtan feladata a szintek közötti összefüggések feltárása. Minimálisan két reprezentációs szintet feltételez, az összetevős szerkezet (c-struktúra) szintjét, amely a *variabilitást* biztosítja, és a funkcionális szerkezet (f-struktúra) szintjét, amely az *univerzalitást* teszi lehetővé. Az elmélet legtöbb változata azonban egyéb szinteket is használ. A szinteket leképezések (mappings) kötik össze, és fontos, hogy ezek a leképezések áttetszőek legyenek (*transzparencia*), amit a nyelvtan monotonitása garantál.

[1] Az ilyen „szigorúan" morféma alapú elméletekről azt is mondhatjuk, hogy a lexikai integritás elvét nem nem tartják be, hanem az számukra semmitmondó, hiszen a szintaxis itt sem lát bele a valós követelményekkel rendelkező lexikai egységek belső szerkezetébe, csak azok jelen esetben nem szavak (amiket tovább lehet még bontani jelentéssel bíró egységekre), hanem tovább már egyébként sem osztható morfémák.

A másik széles körben alkalmazott (megszorítás alapú) grammatika a HPSG (Head-Driven Phrase Structure Grammar). Az explicit megfogalmazás (HPSG-formalizmus) miatt könnyen algoritmizálható, implementálható (emiatt akár számítógépes nyelvészeti irányzatnak is tekinthető). A nevéből következik, hogy szintén használ *frázisstruktúrát* (PSG), a *fejközpontúság* (H) pedig azt jelenti, hogy a függőségi viszonyok a fejbe vannak beépítve. Elutasítja a mag és periféria szétválasztását, minden jelenség feltárása a célja. A HPSG nyelvelméletének objektumai a *lexikon*, az univerzális *elvek*, illetve a nyelvspecifikus elvek és konstrukciók. *Egyszintű*, vagyis nem tekinti a hagyományos nyelvi modulokat különálló komponenseknek, és nem tételez fel különálló reprezentációs formalizmusokat sem az egyes nyelvi szintek – fonológia, morfológia, szintaxis, szemantika – leírására. *Generatív* nyelvtan, vagyis célja egy olyan explicit formális rendszer megadása, amely alkalmas egy adott nyelv jól formált és teljes kifejezéseinek előállítására.

3.2 Gyakorlatban

Az elmúlt évtizedek elméleti sikerei után napjainkban a nyelvtechnológia területén is egyre sikeresebbek a különféle lexikalista elméletek. Mivel céljuk a minél részletesebb elemzés, ezért alkalmasabbak intelligens nyelvtechnológiai célokra, mint a sekélyelemzéssel működő rendszerek. A számítástechnika és a különféle statisztikai módszerek fejlődése lehetővé tette, hogy a nagyobb erőforrásigény ellenére hatékonyak tudjanak lenni az alaposabb nyelvi elemzést végző rendszerek, a megfelelően annotált korpuszok létrehozása pedig a rengeteg szükséges adat rögzítését tudja könnyebbé és gyorsabbá tenni. Az általuk használt unifikációs mechanizmusok használata azért szerencsés, mert így a nyelvtan egyszerűbb szabályokat alkalmaz, gyorsabb, egyszintű, lexikalista és megfordítható (Mitkov 2003), továbbá nem csak a konfigurációs nyelveket (mint az angol) kezeli hatékonyan, hanem a magyarhoz hasonló szabadabb szórendű nyelveket is.

A lexikalista nyelvtanon alapuló alkalmazások közül a legeredményesebbek LFG vagy HPSG formalizmust használnak. Számos nyelvre léteznek már nagy lefedettségű elemzőik, amelyek nagyon jó minőségű és alapos elemzést adnak, továbbá (akár) szemantikai reprezentációt is társítanak a mondatokhoz. Egyik legnagyobb előnyük, hogy különböző nyelvekre alkalmazzák ugyanazt az elméleti keretet, így el tudják érni, hogy a különböző nyelvű elemzések szinte teljesen párhuzamosak legyenek, ami jelentősen megkönnyíti az elemzőkre épülő gépi fordító rendszerek fejlesztését. Az LFG-t mint elméleti keretet használó programok közül a legígéretesebbek azok, amelyeket a Parallel Grammar projekt (ParGram, Butt et al. 2002) keretében fejlesztenek. Nagy lefedettségű elemzőket készítettek már több nyelvre is (például angol, francia, német, japán, urdu), és egyéb nyelvek (köztük a magyar) nyelvtanainak a kidolgozása is folyamatban van. A legkomplexebb HPSG-formalizmust használó alkalmazás pedig a DELPH-IN (Deep Linguistic Processing with HPSG Initiative, pl. Bond et al. 2005), amely nyelvészeti és statisztikai módszerek kombinációját alkalmazza, így az elemzés nem csupán precíz, hanem gyors és robusztus is lehet. Új nyelvtanok kidolgozását a Grammar Matrix nevű alkalmazás segíti (Bender et al. 2002), így több nyelvre fejlesztettek már HPSG alapú grammatikát (például angol, japán, spanyol, francia, görög). Számos nyelvtanhoz szemantikai komponens is tartozik, amelyet az MRS (Minimal Recursion Semantics, Copestake et al. 2005) segítségével valósítanak meg.

4 GeLexi-projekt

4.1 A modell

A lexikalista szemléletet követik a kategoriális nyelvtanok is, amelyeknek egy változata – az unifikációs kategoriális nyelvtan – megvalósítja a *radikális lexikalizmust* (Karttunen 1986), ahol annyira gazdag a lexikai egységek szótári leírása, hogy a frázisstruktúra gyakorlatilag

feleslegessé válik. A mondatok összeépítéséhez két művelet elegendő: a szomszédos egységek kategóriái között működő *függvényalkalmazás* és az egymással relációba kerülő elemek szótári jegyein működő *unifikáció*. Az unifikációs mechanizmus az attribútum–érték párokba rendezett morfológiai jegyek ellenőrzéséért felelős, kiszűri a hibás egyeztetéseket, eseteket stb. Az elemzést a kategoriális nyelvtan függvényalkalmazás nevű művelete végzi, amely összerakja a mondat egymás melletti szavait, melléktermékként létrehozva egy közvetlen összetevős szerkezeti fát, amely csupán „analízis-fa", a róla leolvasható összetevőknek nincsen semmi nyelvészeti relevanciája. Fellép továbbá egy nagyfokú „hamis többértelműség" a legtöbb mondat esetében, mivel ugyanahhoz az olvasathoz számtalan eltérő analízis-fa tartozik, egyetlen unifikálódott morfológiai jegyrendszer mellett. „All that matters is the resulting feature set" [csakis az adódó jegyhalmaz számít, semmi más] – állapítja meg Karttunen, és azt a végkövetkeztetést vonja le, hogy egy *radikálisan lexikalista* megközelítés eredményesebb volna, az eddigiek mellett számítástechnikai szempontból is.

A GASG (Generative/Generalized Argument Structure Grammar, Alberti 1999) egy olyan módosított unifikációs kategoriális nyelvtannak tekinthető, amely megvalósítja ezt az elvet. A kiüresedett nyelvészeti tartalmú szintaktikai összerakó-művelet teljes kiküszöbölésén alapul: a már eleve redukált szintaxisból kitöröljük még a függvényalkalmazás műveletét is. Nem épülnek fák, ezáltal a nyelvtan mentes a fent említett hamis többértelműségtől is. Ami marad, az csupán az *unifikáció* művelete. A szórendről is ennek segítségével ad számot: egy szó szomszédossági követelménye ugyanúgy egy unifikálandó jegy, mint a száma, vagy hogy milyen esetű argumentumot vár. Az elmélet egy nagyon erős univerzális grammatikai megszorítást tesztel: azt vizsgálja, hogy lehetséges-e úgy leírni a nyelvet, hogy nem támaszkodunk szintaktikai szabályrendszerre. Tehát nem csupán a transzformációk létét tagadja, hanem frázisstruktúrát sem épít; a grammatikalitásról és a mondatok jelentéséről pusztán a gazdagon strukturált lexikon és az unifikáció művelete gondoskodik.

Számítógépes nyelvészeti munkákban is (pl. Schneider 2005) találkozhatunk azzal a nézettel, hogy a frázisstruktúra redukálása hasznos lehet, sok alkalmazás pusztán a hatékonyság növelése miatt támaszkodik rá, mert nélküle függőségi nyelvtant kapunk, amely – mivel nem tartalmaz a szórendre vonatkozó megszorításokat – számítástechnikai szempontból nem hatékony. A GASG azonban számot ad a szórendről is, így a frázisstruktúra feladása nem feltétlenül eredményez exponenciális futásidejű elemző algoritmust.

A GASG létjogosultsága melletti egyik fő elméleti érv, hogy *kompozicionális* szemantikai partnereként szolgálhat a DRT-szerű diskurzus-szemantikai elméleteknek (DRT: Discourse Representation Theroy, van Eijck–Kamp 1999). A Frege-féle kompozicionalitási elv értelmében a kutatók régóta keresnek egy izomorf szintaxis–szemantika párt. Montague szolgáltatott először (a '70-es években) egy olyan szemantikaelméletet, amely a chomskyánus szintaxissal kompozicionális tudott lenni, amihez azonban (a két rendszer közötti nagymértékű különbözőség miatt) egy nagyon erős eszköz (a λ–absztrakció) alkalmazására volt szükség. Egy másik lehetséges megoldása a problémának, ha egy jól működő szemantika-elméletet tekintünk kiindulásnak, és ahhoz keresünk egy vele kompozicionális szintaxist; ezt valósítja meg a GASG, amelyre ugyanaz a hierarchiamentesség jellemző, mint a DRT-re, így a két rendszer között egy szigorúbb értelemben vett kompozicionalitás tud megvalósulni.

A nyelvtan tehát nem más, mint egy óriási lexikon, ahol a lexikai egységek jellemzése tartalmazza az elem saját tulajdonságait (fonológiai forma, szófaj, vonzatkeret, az egységhez tartozó proto-DRS stb.), valamint a „környezeti követelményeit", vagyis hogy hány és milyen tulajdonságú elemet keres a mondatban. A szórendre vonatkozó megszorítások ugyanúgy egy lexikai tulajdonságként vannak tárolva, mint például az egyeztetéssel kapcsolatos elvárások, ezért nincs szükség frázisstruktúra építésére. A GASG tehát egy *homogén* nyelvtan, amely egységesen kezeli a különféle nyelvi jelenségeket, mint például a szavak jólformáltsága, morfémasorrend, a vonzatok megléte, eset, egyeztetés, szórend, szemantikai jegyek, vagy

akár szemantikai reprezentáció hozzárendelése: csak tulajdonságok és követelmények vannak, amelyek ha tudnak unifikálódni, grammatikus lesz a mondat, és előállnak a különféle reprezentációk. Ezáltal nagyfokú univerzalitás érhető el: bármely nyelv leírását ugyanabban a keretben lehet megtenni, attól függetlenül, hogy az adott nyelv milyen eszközökkel (szórend vagy morfológia) fejezi inkább ki a grammatikai viszonyokat; az elmélet nem tulajdonít nagyobb jelentőséget egyik módszernek sem a másikhoz képest.

A szintaktikai elemzés eredménye – mivel frázisstruktúra nem épül – gyakorlatilag egy függőségi viszonyrendszer; viszont az elmélet nem csupán egy függőségi nyelvtan, mert számot ad a szórendről is: egyrészt a különböző szórendi variánsok grammatikalitásáról, másrészt az eltérő szórenddel járó szemantikai változásokról (például fókusz). Mindezt az ún. *rangparaméterek* segítségével éri el, amelyek azt rögzítik, hogy egy adott követelmény milyen erősségű. Például a szórend esetében a kiindulás az, hogy minden elem szomszédos kíván lenni minden olyan elemmel, amellyel szintaktikai viszonyt létesít, ami gyakran nem tud teljesülni, hiszen egy elemmel több másik is kapcsolatot létesíthet, de legfeljebb kettő tud vele szomszédos lenni. Ezért van szükség arra, hogy ez a közvetlen megelőzési igény rangparaméterekkel legyen ellátva, vagyis rögzítve legyen, melyik igény milyen erősségű, és ha ellentmondó követelmények lépnek fel, akkor az erősebb „győz”. A rangparaméterek alkalmazása kiterjeszthető más jelenségekre is, például a fókusz vagy a progresszivitás kezelésére; illetve segítségével még nagyobb fokú univerzalitás érhető el, hiszen elvileg bármilyen tulajdonság vagy elvárás mellé rögzíthető rangparaméter, amelyek egymással „harcolva” végül kiadják az adott nyelv adott jelenségének felszíni megvalósulását az adott mondatban (az optimalitáselmélet filozófiájához hasonlóan).

Az elmúlt évtizedekben megfigyelhető erőteljes lexikalista fordulat „végső” állomása a totális lexikalizmus, ahol már frázisstruktúra sem épül, és nincsenek szintaktikai műveletek, a mondatok pusztán *unifikáció* segítségével épülnek össze. Nincsenek „nagy szabályok”, minden követelmény *lokális*, amelyek együttes kielégülése eredményezi a grammatikus mondatokat (*deklarativitás, egyszintűség*). A *szemantika* a központi komponens, a szintaxis elsődlegesen azt a célt szolgálja, hogy jelentéstani reprezentációt tudjunk a mondatokhoz társítani. Kiindulópontja a nem hierarchikus felépítésű diskurzusreprezentációs struktúra (DRS), amihez a totálisan lexikalista GASG egy kompatibilis (szintén nem hierarchikus) szintaxist kínál.

Nem állítjuk, hogy az így nyert teljesen homogén (csak lexikonból álló) és hierarchiamentes megközelítés feltétlenül eredményesebb, mint az egyéb (nagyon sikeres) lexikalista elméletek; azonban több okból is érdemes kipróbálni. Először is elméletileg érdekes kérdés, hogy hasznos-e a bizonyítottan jól működő lexikalizmust a végsőkig fokozni; illetve hogy megfelelően hatékony tud-e lenni a gyakorlatban egy elméleti szempontból feltétlenül elegánsabb homogén nyelvtan, vagy a nyelv olyan, hogy szótárra és szintaktikai szabályokra egyaránt hivatkozik (Pinker 1999). Gyakorlati szempontból pedig azért tűnik ígéretesnek a megközelítés, mert minden eddiginél univerzálisabb keretet biztosít a nyelvek leírásához – ebből a szempontból az optimalitáselmélettel rokonítható – azáltal, hogy minden nyelvben ugyanazokat a megszorításokat tételezi fel, csak a hozzájuk rendelt rangparaméterek térhetnek el, ami a nyelvek közötti változatosságot eredményezi; illetve hogy minden eddiginél közvetlenebb módon képes szemantikai reprezentációt társítani a mondatokhoz, hiszen nem kell feloldania egy hierarchikus szintaxis és egy „lapos” szemantika közötti ellentmondást.

4.2 Az implementáció

Nem csupán gyakorlati, hanem elméleti haszna is van, ha egy nyelvelmélethez implementáció készíthető, hiszen a működő algoritmusok teszik kétségtelenné, hogy jól formalizált, egzakt rendszer áll rendelkezésünkre. Alapvetően emiatt döntöttünk úgy, hogy elkészítjük a totálisan

lexikalista GASG számítógépes implementációját. További célunk megvizsgálni a totális lexikalizmus használhatóságát a nyelvtechnológiában, hogy alapjain lehetséges-e jól működő nyelvelemzőt készíteni, és ha igen, az kellően hatékony tud-e lenni. Az implementáló programunk tevékenysége nem más, mint a „generatív alapfeladat" végrehajtása: egyértelműen el kell döntenie egy beírt szósorról, hogy az grammatikus-e, és amennyiben az, morfoszintaktikai és szemantikai reprezentációt kell hozzá rendelnie. Ezek alapján a gyakorlati célunk egy mondatelemző létrehozása, amely kezdetben magyar, majd angol, illetve egyéb nyelvű szövegeket tud elemezni, hozzájuk reprezentációkat társítani, és köztük a gépi fordítást megvalósítani. A kiinduló (elméleti) cél megvalósítására alakult a GeLexi-projekt, a gyakorlatibb (hatékonyságot is számításba vevő) célra pedig a LiLe-projekt, majd a még alaposabb szemantikai reprezentációt célzó ReALIS-projekt.

Az implementáció készítése visszahathat magára az elméletre is, hiszen felszínre kerülhetnek olyan problémák, amelyeket az elmélet eredeti formájában nem tudna kezelni, illetve születhetnek olyan megoldások, amelyek hatékonyabban kezelik a nyelvet, mint az eredeti elképzelés, de az elmélet szellemiségével nincsenek ellentmondásban, vagy akár még jobban megvalósítják azt. Ez utóbbi történt a GASG implementálása során is, amikor egyértelművé vált, hogy a nyelvtan kívánatos teljes homogenitását az biztosítja, ha a morfológiát is totálisan lexikalista módon közelítjük meg: nem a szavakhoz, hanem közvetlenül a morfémákhoz rendeljük a gazdagon strukturált, minden grammatikai szintről egyidejűleg információt hordozó lexikai egységeket, és a mondatelemzés során az is eldől, hogy mely elemek épülnek össze szóvá, és melyek alkotnak (egymás közelében maradó) külön szavakat. A totálisan lexikalista morfológia számára tehát nem okoz gondot, hogy a különböző nyelvekben különböző helyen található a szószint, hiszen az is csak egy tulajdonság, hogy az adott lexikai egység (állítást hordozó morféma) külön szót alkot-e, vagy affixumként valósul meg.

A GeLexi-projekt célja tehát annak igazolása volt, hogy a GASG egy jól formalizált, egzakt rendszer, illetve hogy a totálisan lexikalista megközelítés alkalmazható a számítógépes nyelvészetben, eldönthető egy mondatról annak grammatikus volta csupán a lexikonban tárolt jegyek unifikációja alapján. Az elmélet implementációját Prolog programnyelven készítettük, amelyet kimondottan arra a célra fejlesztettek, hogy unifikációval tudjon különféle problémákat megoldani; így tehát a GASG által alkalmazott egyetlen műveletet beépítve tartalmazza. Az adatbázisba csupán néhányszáz lexikai egységet rögzítettünk, hiszen a célunk (első lépésként) a mechanizmusok működőképességének vizsgálata volt, a hatékonyság kérdését egyelőre félretettük.

A program bemenete egy magyar vagy angol nyelvű szósor, amelyről az elemző eldönti, hogy grammatikus mondatot alkot-e, és amennyiben igen, négyféle reprezentációt társít hozzá. Az első kimenet a releváns lexikai egységek listáját tartalmazza, ezt követi a szintaktikai viszonyrendszer ábrázolása, hogy mely lexikai egységek között milyen grammatikai reláció létesül, majd a mondat ún. kopredikációs hálózatát olvashatjuk, vagyis hogy mely elemek tesznek állítást ugyanarról a referensről, végül pedig a szemantikai reprezentáció látható, egy LDRS (Lifelong DRS, Alberti 2000). Az implementálás során többféle nyelvi jelenség kezelését kidolgoztuk: a program számot ad a morfofonológiai váltakozásokról, a zéró névmásokról, elemezni tud műveltetést, igekötőt vagy vonzatos melléknevet tartalmazó mondatot, illetve a mellérendelés kidolgozását is megkezdtük.

A szemantikai reprezentáció egy egyszerűsített angol nyelven tartalmazza a mondatban található összes információt, így rendszerünk a géppel támogatott fordítás területén is hasznosítható. Míg egy külföldinek évekbe telne megtanulnia magyarul, addig a program által előállított DRS-szerű szemantikai reprezentáció olvasását bármely angolul értő felhasználó alig egy óra alatt elsajátíthatja. A ragozó nyelvek esetében különösen hasznos lehet egy ilyen eszköz – szemben egy egyszerű szótárral –, hiszen például a *Kerestelek* mondat fordításakor

nem található meg a szótárban az alany és a tárgy számára és személyére vonatkozó információ, ahogy az sem, hogy a cselekvés a múltban zajlott.

Angol nyelvű lexikai egységeket azért vettünk fel az adatbázisba, hogy megmutassuk a rendszer univerzalitását, vagyis hogy a struktúra más nyelvek lexikai egységeinek a tárolására is alkalmas, amelyek lexikonban tárolt jegyein ugyanúgy működik az unifikáció. A program tehát angol nyelvű mondatokat is képes elemezni, és ugyanolyan típusú reprezentációkat társít hozzájuk, mint a magyar nyelvűekhez. A nyelvek közötti különbségek jelentős része már a kopredikációs hálózat szintjén eltűnik, egy magyar nyelvű mondathoz és annak angol megfelelőjéhez rendelt reprezentáció csak az elemek számozásában tér el.

Kihasználva, hogy amit a Prolog elemezni tud, azt generálni is, a program kétirányú használatával a gépi fordítást is megvalósítottuk a két nyelv között (továbbra is totálisan lexikalista alapokon). A rendszer először elemzi a forrásnyelvi mondatot – közben előállítja a különféle reprezentációkat –, azután a kopredikációs hálózat alapján betölti a célnyelvi lexikai egységeket, majd (az egyeztetésért felelős morfémák helyére változókat feltételezve) generálja a lehetséges morfémasorokat, végül ezekre meghívja a célnyelvi elemzőt, így a fordítás kimenete csak grammatikus mondat lehet. A kidolgozott keret univerzális, így a program elvileg bármely két nyelv között meg tudná valósítani a fordítást, amelyek lexikai egységeit tartalmazná. A rendszer számára a nagyon különböző szerkezetű nyelvek közötti fordítás sem nehezebb, mivel minden állítással bíró elem egy-egy lexikai egység (totálisan lexikalista morfológia), így nem okoz gondot a szószint esetleges különbözősége; illetve minden nyelvi tulajdonság, minden elvárás és minden „szabály” (még a sorrendre vonatkozó megszorítások is) az adott lexikai egység leírásában kerülnek tárolásra, vagyis minden információ a lexikonban található, így nincs szükség nyelvspecifikus szabályokra.

4.3 Néhány programfutás

A program működését egy műveltetést tartalmazó mondaton mutatom be. A műveltetés egy olyan nyelvi jelenség, amelyet a különböző nyelvek nagyon különböző módokon tudnak kifejezni. A magyar nyelv képzőt használ erre a célra, míg az angol önálló szavakat (*make, have*). A morféma alapú totálisan lexikalista megközelítés egységesen tudja ezt kezelni, hiszen mindegyik esetben az adott morféma egy önálló lexikai egysége az adott nyelvnek, és irreleváns, hogy toldalékként vagy szóként jelenik-e meg. A nagyfokú különbözőség fordításkor sem okoz problémát, hiszen nem csupán a szemantika, hanem már a kopredikáció szintjén sem látható ez a felszíni különbség (a példa ezt is érzékelteti). A mondat a műveltetésen túl tartalmaz modalitásért felelős elemet is, amelyet szintén eltérően fejez ki az angol és a magyar nyelv (önálló szó vs. képző). (1)-ben látható a célállítás, amelyben azt kérjük a programtól, hogy fordítsa le a *Péter énekeltetheti Marit* mondatot, és írja ki az elemzéseket is. (2)-ben olvasható az első reprezentáció, amely a lexikai egységek listája.

```
(1)  translate_Hun_Eng_print("Péter énekeltetheti Marit.").

(2)  LEXIKAI EGYSÉGEK:

     Péter:  n(1,1,li(m("","Péter",""),labstem("Peter",phonfst(1,2,0,2),1,[])))

     énekel:  n(2,1,li(m("","énekel",""),labstem("sing",phonfst(1,2,2,2),2,[["NOM"]])))
     tet:  n(2,2,li(m("t","A","t"),labder("cause",phonfsu(2,2,0.2,2),2,ac(-1,0,1))))
     het:  n(2,3,li(m("h","A","t"),labsuff("may",phonfsu(1,1,1,2),2,1)))
     i:  n(2,4,li(m("","i",""),labsuff("sg3obj+def",phonfsu(1,3,1,3),2,3)))

     Mari:  n(3,1,li(m("","Mari",""),labstem("Mary",phonfst(2,2,0,2),1,[])))
     t:  n(3,2,li(m("V","t",""),labsuff("ACC",phonfsu(1,1,1,3),1,4)))
```

Mindegyik sor elején az adott allomorf olvasható, majd pedig a releváns lexikai egység, amelyet a program az adatbázisból betöltött. Először egy számozás látható (hányadik szó

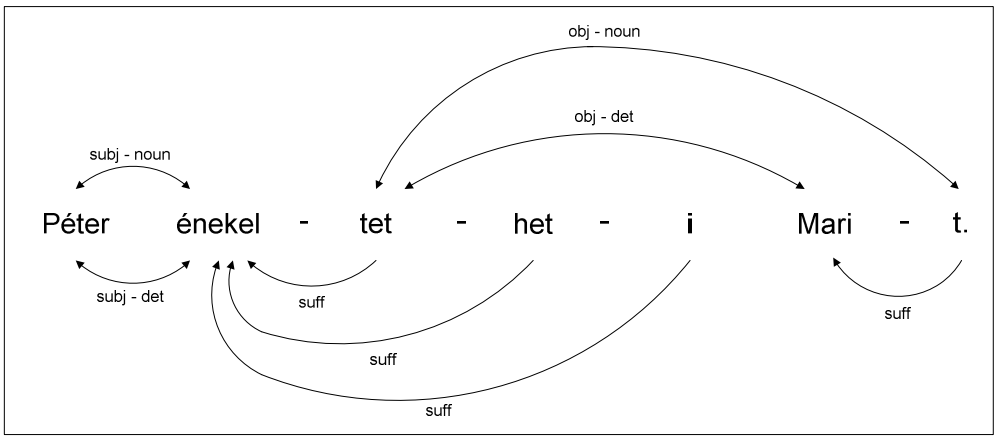
hányadik morfémája az adott elem), majd az egységhez tartozó *sajátszó* (hangalak, amely változókat is tartalmazhat), végül a hozzá tartozó címke a predikátum angol nyelvű megnevezésével és az egység különféle tulajdonságaival. Például az utolsó elem (a harmadik szó második morfémája) az ebben a mondatban *-t*-ként megvalósuló *-Vt* sajáttestű tárgyesetet jelző lexikai egység, ahol a *V* változó lehetséges értékei: a, e, o, ö és üres. A fonológiai jegyei: 1, 1, 1, 3, vagyis nyújt (*kutya–kutyát*), rövidít (*madár–madarat*), kivet (*bokor–bokrot*), a nyitás (*botot–botokat*) viszont nem releváns, mivel utána nem állhat már toldalék. Morfológiai jegyei 1 és 4, vagyis főnév és 4-es ranggal akar a tőhöz kapcsolódni (ami egy gyenge követelmény, vagyis a szó végén fog elhelyezkedni).

(3)-ban a második reprezentációs szint – a szintaktikai viszonyrendszer – látható, először egy részletesebb formában (morfémák közötti relációk, ezt mutatja az 1. ábra is), majd egy áttekinthetőbb (egyszerűsített) formában, ahol a szavak közötti viszonyok olvashatók.

(3) SZINTAXIS:

```
gr("noun","regent","subj",1,1,2,1)
gr("det","regent","_",1,1,2,1)
gr("regent","noun","subj",2,1,1,1)
gr("regent","det","subj",2,1,1,1)
gr("suff","stem","free",2,2,2,1)
gr("regent","noun","obj",2,2,3,2)
gr("regent","det","obj",2,2,3,1)
gr("suff","stem","free",2,3,2,1)
gr("suff","stem","free",2,4,2,1)
gr("det","regent","_",3,1,2,1)
gr("suff","stem","free",3,2,3,1)
gr("noun","regent","obj",3,2,2,1)
```

```
regent-noun-subj: énekeltetheti-Péter
regent-det-subj: énekeltetheti-Péter
regent-noun-obj: énekeltetheti-Marit
regent-det-obj: énekeltetheti-Marit
```



1. ábra: Szintaktikai viszonyok

A következő reprezentáció (4) a kopredikációs viszonyok hálózatát mutatja (vagyis hogy mely elemek tesznek ugyanarról állítást), amely a gépi fordításhoz hasznos.

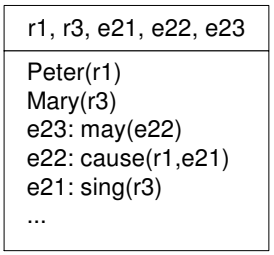
(4) KOPREDIKÁCIÓS VISZONYOK:

```
copr("sing",2,1,"Mary",3,1,1,1,"arg")
copr("sing",2,1,"Mary",3,1,1,0,"arg")
copr("cause",2,2,"Peter",1,1,1,1,"arg")
copr("cause",2,2,"Peter",1,1,1,0,"arg")
copr("cause",2,2,"sing",2,1,2,0,"arg")
copr("may",2,3,"cause",2,2,1,0,"arg")
```

Végül az utolsó (és legfontosabb) reprezentáció a szemantika (5), ami egy (L)DRS. A mellette lévő ábrán láthatók a fő predikátumai, vagyis hogy két entitás szerepel: r1 (Péter) és r3 (Mari); illetve három eseményre történik utalás: e23 az, hogy lehetséges e22, amely szerint r1 (Péter) e21-et okozza, vagyis hogy r3 (Mari) énekel. A reprezentációban egyéb (a DRS kontextusba ágyazásához szükséges) állítások is olvashatók.

(5) SZEMANTIKA:

```
provref("fixpoint",[e(2,3,1)])
provref("old",[r(1,1,1)])
pred("Peter",1,[r(1,1,1)])
provref("new",[e(2,1,1)])
pred("sing",2,[e(2,1,1),r(3,1,1)])
provref("new",[e(2,2,1)])
provref("=",[e(2,2,1),e(2,1,1)])
pred("cause",2,[e(2,2,1),r(1,1,1),e(2,1,1)])
provref("new",[e(2,3,1)])
provref("<",[e(2,3,1),e(2,2,1)])
pred("may",2,[e(2,3,1),e(2,2,1)])
provref("old",[r(3,1,1)])
pred("Mary",3,[r(3,1,1)])
```



A magyar nyelvű mondat elemzése után a program generálja a célnyelvi (angol) mondatot, és amikor megtalálta a megfelelő fordítást, kiírja az (angol nyelvű) elemzésekkel együtt (6).

(6) In English: Peter may make Mary sing.

LEXICAL ITEMS:

```
Peter:  n(1,1,li(m("","Peter",""),labsteme("Peter",1,[["0"]])))
may:  n(2,1,li(m("","may",""),labsteme("may",2,[["VERB"]])))
make:  n(3,1,li(m("","make",""),labsteme("cause",2,[["NOM","VERB"]])))
Mary:  n(4,1,li(m("","Mary",""),labsteme("Mary",1,[["0"]])))
sing:  n(5,1,li(m("","sing",""),labsteme("sing",2,[["NOM"]])))
```

SYNTAX:

```
gr("noun","regent","subj",1,1,3,1)
gr("det","regent","_",1,1,3,1)
gr("regent","verb","arg",2,1,3,1)
gr("regent","noun","subj",3,1,1,1)
gr("regent","det","subj",3,1,1,1)
gr("regent","verb","arg",3,1,5,1)
gr("noun","regent","subj",4,1,5,1)
gr("det","regent","_",4,1,5,1)
gr("regent","noun","subj",5,1,4,1)
gr("regent","det","subj",5,1,4,1)
```

```
regent-verb-arg: may-make
regent-noun-subj: make-Peter
regent-det-subj: make-Peter
regent-verb-arg: make-sing
regent-noun-subj: sing-Mary
regent-det-subj: sing-Mary
```

COPREDICATIVE NETWORK:

```
copr("may",2,1,"cause",3,1,1,0,"arg")
copr("cause",3,1,"Peter",1,1,1,1,"arg")
copr("cause",3,1,"Peter",1,1,1,0,"arg")
copr("cause",3,1,"sing",5,1,2,0,"arg")
copr("sing",5,1,"Mary",4,1,1,1,"arg")
copr("sing",5,1,"Mary",4,1,1,0,"arg")
```

SEMANTICS:

```
provref("fixpoint",[e(2,1,1)])
```

```
provref("old",[r(1,1,1)])
pred("Peter",1,[r(1,1,1)])
provref("new",[e(2,1,1)])
provref("<",[e(2,1,1),e(3,1,1)])
pred("may",2,[e(2,1,1),e(3,1,1)])
provref("new",[e(3,1,1)])
provref("=",[e(3,1,1),e(5,1,1)])
pred("cause",3,[e(3,1,1),r(1,1,1),e(5,1,1)])
provref("old",[r(4,1,1)])
pred("Mary",4,[r(4,1,1)])
provref("new",[e(5,1,1)])
pred("sing",5,[e(5,1,1),r(4,1,1)])

yes
```

Megfigyelhető, hogy a két nyelv kopredikációs hálózata a mondatok közötti szerkezetbeli eltérések ellenére is nagyon hasonló: pusztán az elemek sorszámában, illetve a felsorolás sorrendjében térnek el egymástól. Természetesen a szemantikai reprezentációk is egyformák.

A program hasznosságát a géppel segített fordításban talán a zéró névmásokat tartalmazó mondatok elemzésén lehet leginkább bemutatni. Hiszen ha a magyar mondat minden állítást egy-egy (önálló) szóval fejez ki, akkor a program nem nyújt sokkal többet, mint egy kétnyelvű szótár. Ha azonban egy magyarul nem tudó külföldi azt a mondatot látja, hogy *Szeretlek*, arra talán rájön egy szótár segítségével, hogy a 'love' predikátum szerepel benne, azt viszont nem fogja tudni kikövetkeztetni, hogy ki szeret kit. Ennek oka, hogy a magyar ún. pro-drop nyelv, azaz a névmások sokszor elhagyhatók, és csak a ragozásból tudunk a szereplők kilétére következtetni. Ha viszont beírja ezt a mondatot a programba, megkapja a szemantikai reprezentációt, ahol olvashatja, hogy aki szeret, az r011 (0: beépített, mindig jelenlévő referens, 11: egyes szám első személyű, vagyis 'én'), akit szeret, az pedig r012 (egyes szám második személyű névmás, vagyis 'te'). (7) a mondat fordítását mutatja, ahol először az elemzését olvashatjuk, amelynek legfontosabb sora a szemantikai reprezentációban olvasható (félkövérrel szedve), majd az angol fordítást, végül annak az elemzését.

(7) `translate_Hun_Eng_print("Szeretlek").`

```
LEXIKAI EGYSÉGEK:

szeret:  n(1,1,li(m("","szeret",""),labstem("love",
                                        phonfst(1,2,2,2),2,[["NOM","ACC"]])))
l: n(1,2,li(m("","l",""),labsuff("objperson2",phonfsu(3,2,1,1),2,2.5)))
ek: n(1,3,li(m("V","k",""),labsuff("sg1",phonfsu(1,1,2,3),2,3)))

SZINTAXIS:

gr("suff","stem","free",1,2,1,1)
gr("suff","stem","free",1,3,1,1)

KOPREDIKÁCIÓS VISZONYOK:

copr("love",1,1,"sg1",1,3,1,1,"arg")
copr("love",1,1,"objperson2",1,2,2,1,"arg")

SZEMANTIKA:

provref("fixpoint",[e(1,1,1)])
provref("new",[e(1,1,1)])
pred("love",1,[e(1,1,1),r(0,1,1),r(0,1,2)])

In English:  I love you.

LEXICAL ITEMS:

I:  n(1,1,li(m("","I",""),labsteme("I",1,[["0","sg","1","NOM"]])))
love:  n(2,1,li(m("","love",""),labsteme("love",2,[["NOM","ACC"]])))
you:  n(3,1,li(m("","you",""),labsteme("you",1,[["0","_","2","_"]])))
```

```
SYNTAX:

gr("noun","regent","subj",1,1,2,1)
gr("det","regent","_",1,1,2,1)
gr("regent","noun","subj",2,1,1,1)
gr("regent","det","subj",2,1,1,1)
gr("regent","noun","obj",2,1,3,1)
gr("regent","det","obj",2,1,3,1)
gr("noun","regent","obj",3,1,2,1)
gr("det","regent","_",3,1,2,1)

regent-noun-subj: love-I
regent-det-subj: love-I
regent-noun-obj: love-you
regent-det-obj: love-you

COPREDICATIVE NETWORK:

copr("love",2,1,"I",1,1,1,1,"arg")
copr("love",2,1,"I",1,1,1,0,"arg")
copr("love",2,1,"you",3,1,2,1,"arg")
copr("love",2,1,"you",3,1,2,0,"arg")

SEMANTICS:

provref("fixpoint",[e(2,1,1)])
provref("old",[r(1,1,1)])
pred("=",1,[r(1,1,1),r(0,1,1)])
provref("new",[e(2,1,1)])
pred("love",2,[e(2,1,1),r(1,1,1),r(3,1,1)])
provref("old",[r(3,1,1)])
pred("you",3,[r(3,1,1)])

yes
```

Ha egy mondatnak több célnyelvi megfelelője is van, a program ki tudja írni az összes lehetséges megoldást (a *fail* parancs segítségével). (8)-ban az látható, hogy az *I love you* angol mondatnak elvileg hat magyarra fordítása lehetséges, amelyek közül az első (ami azt is jelenti, hogy elsődleges, vagyis ha nincs *fail* parancs, az egyetlen) a *Szeretlek* mondat, vagyis amikor a névmásokat nem tesszük ki. Ez a legsemlegesebb, a többi esetben valami (minimálisan pragmatikai) pluszjelentést hordoz a magyar mondat (topik, fókusz, amelyeket egyelőre nem kezel a program).

```
(8)  translate_Eng_Hun("I love you."),fail.

     In Hungarian: Szeretlek.
     In Hungarian: Szeretlek téged.
     In Hungarian: Szeretlek titeket.
     In Hungarian: Én szeretlek.
     In Hungarian: Én szeretlek téged.
     In Hungarian: Én szeretlek titeket.
     no
```

Programunk több téren is újdonságot nyújt. A legfontosabb, hogy működő szemantikai komponenst tartalmaz, és ténylegesen képes a beírt mondatokhoz modern szemantikai reprezentációt társítani. A megközelítésünk alapjául szolgáló totális lexikalizmusnak elméleti és gyakorlati előnyei is vannak. Elméleti előny a homogenitás, azaz nincs külön szintaxis és lexikon, csak lexikon van[2]. Elméleti és gyakorlati előny egyben maga a lexikalizmus, amellyel a szabadabb szórendű nyelvek (mint például a magyar is) könnyebben kezelhetők. Programunk tisztán gyakorlati előnye pedig a számítástechnikában manapság kívánatos

[2] Egyéb homogén rendszerek is léteznek, amelyek azonban inkább a lexikont száműzik, és csak szintaktikai szabályokkal dolgoznak (pl. Prószéky et al. 2004). Azonban a nyelvészetben az utóbbi években megfigyelhető erősen lexikalista fordulat inkább a mi megközelítésünket igazolja, legalábbis elméleti szemszögből.

„minimális processzálás – maximális adattár”. Végül pedig a fordítás totálisan lexikalista megközelítésének előnye, hogy univerzális tud lenni a keret, amelyet használ, nem pedig nyelvspecifikus, ezért nem kell külön-külön kidolgozni minden egyes nyelvpárra a fordítás mechanizmusát. Amint a nyelvek elemzői rendelkezésünkre állnak, bármely nyelvről fordítani tudunk a másikra. Ezért is van, hogy egyidejűleg működik programunkban a magyarról angol nyelvre, illetve az angolról magyar nyelvre történő fordítás.

A GeLexi-projekt által készített elemző tehát igazolta, hogy a totálisan lexikalista megközelítés eredményes lehet, a mechanizmusok működnek, és valóban nincs szükség frázisstruktúra építésére a mondatok elemzéséhez. A mondatok szórendjéről egy sokkal egységesebb eszközzel (a rangparaméterekkel) is számot lehet adni, amely ugyanolyan követelmény, mint hogy milyen esetű argumentumot vár egy régens. Az így nyert nem hierarchikus szintaxis közvetlenül kompozicionális a szemantikával, így a szemantikai reprezentáció egyszerűbben előállítható, mint a frázisstruktúrát használó elméletek esetében.

5 LiLe-projekt

A GeLexi-projekt tehát egy működő implementációval igazolta, hogy a GASG egy jól formalizált, egzakt rendszer, és hogy a totális lexikalizmus alkalmazható a nyelvtechnológia területén. Következő lépésként a LiLe-projekt vállalkozott arra, hogy a rendszert egy modernebb fejlesztőkörnyezetbe helyezi, ahol már a hatékonyság is vizsgálható, és a felhasználóbarát felület is fontos szempont.

A lexikon számára egy SQL-adatbázist készítettünk, amelynek a tartalma weben keresztül is megjeleníthető, és hozzá Delphi programnyelven feltöltő- és elemzőprogramot írtunk; így a modernebb környezetbe való helyezést és a felhasználóbarát felület létrehozását teljesítettük. A megvalósításban a morfológiai komponensig jutottunk, az azzal kapcsolatos táblákat töltöttük fel adatokkal, így a program is csupán morfológiai szempontból tudja elemezni a beírt szavakat.

A program egy beírt szóról el tudja dönteni, hogy jól formált-e, és morfológiai elemzést tud hozzá társítani. Két szempontból több, mint egy hagyományos morfológiai elemző: egyrészt a szabályok kikapcsolhatók, vagyis kérhetjük a programot, hogy például a hangrendi illeszkedésre vagy a morfémák sorrendjére vonatkozó megszorításokat ne vegye figyelembe; másrészt ha a szóalak nem bizonyult jól formáltnak, kiírja azt is, mi volt ennek az oka, milyen szabályt sért. A 2. ábra a *lovakat* szóalak elemzését mutatja. Az első sorban láthatjuk a jó megoldást, a további sorokban pedig az alak egyéb lehetséges szegmentálásait, mindegyik esetben feltüntetve, miért nem lehetséges az adott elemzés (az adott lexikai egység mely jegyei nem tudtak unifikálódni).

ló; többesszám jele; tárgyrag; = OK
ló; többesszám jele; Poss E/3; múlt idő jele;
= Nem egyezik a morfémák szófaja: [Poss E/3, névszó] -> [múlt idő jele, ige]
ló; többesszám jele; Poss E/3; tárgyrag;
= Nem jó a morfémasorrend: [többesszám jele] -> [Poss E/3]
ló; Poss E/3; többesszám jele; tárgyrag;
= Nem jó a morfémasorrend: [Poss E/3] -> [többesszám jele]
ló; Poss E/3; többesszám jele; Poss E/3; múlt idő jele;
= Nem egyezik a morfémák szófaja: [Poss E/3, névszó] -> [múlt idő jele, ige]
ló; Poss E/3; többesszám jele; Poss E/3; tárgyrag;
= Nem jó a morfémasorrend: [Poss E/3] -> [többesszám jele]

2. ábra: A *lovakat* szóalak elemzése

A LiLe-projekt lexikona számos pozitív tulajdonsággal rendelkezik. Több információt tartalmaz a lexikai egységekről, mint akármilyen másik szótár, ami nagyon hasznos bármilyen adatbázisnál. A struktúra nyelvfüggetlen, így egyszer kell csak kidolgozni, később (amikor

más nyelvek kerülnek bevonásra) csak újabb rekordokat (lexikai egységeket és „szabályokat”) kell felvinni. Végül pedig a szabályok ki-bekapcsolhatók, ami nem csupán az oktatásban hasznos, hanem lehetővé teszi, hogy ne csak a tökéletesen grammatikus alakokat ismerje fel a program (Prószéky (2005) érvel például az ilyen rendszerek szükségessége mellett).

A projekt eredetileg arra a célra alakult, hogy a GeLexi-projekt eredményeit egy modernebb fejlesztőkörnyezetben valósítsa meg, és ezáltal tesztelhessük a totálisan lexikalista megközelítés hatékonyságát nagyobb adatbázison. Ezt a célt csak részben érte el: létrehoztuk ugyan az adatbázist, amely a feltöltött adatokon jól működik, viszont nem növeltük jelentősen a lexikon méretét, illetve a megvalósításban csak a morfofonológia szintjéig jutottunk. A munka azért szakadt meg, mert a két projekt egyesülésével új adatbázison dolgozunk; kicsit más alapokon és kicsit más célokat helyezve előtérbe (ReALIS-projekt).

A LiLe-projekt jelentősége a totális lexikalizmus legitimálásában az, hogy kidolgozta a lexikai egységek SQL-adatbázisban való tárolásának módját, létrehozott egy olyan struktúrát, amely a későbbi kutatásokhoz alapul szolgálhat; illetve újabb (közbeeső) célokat fogalmazott meg, mint például az oktatás támogatása, vagy egy „dinamikus korpusz” létrehozása (amelyben nem a létező, hanem az elvileg lehetséges alakok kereshetők). A morfofonológiai komponensnek nem csupán a táblastruktúráját, hanem a bevitt adatokat (szófajok, fonológiai tulajdonságok stb.) is be tudtuk építeni az új alkalmazásba, hiszen ezek működőképességét már bizonyítottuk. Továbbá ebben a munkában jelent meg először a totális lexikalizmus eszméjébe maximálisan illeszkedő lehetőség, hogy bármely unifikálandó jegy (tulajdonság vagy elvárás) kikapcsolható, ami a rendszer robusztusságát tudja biztosítani.

6 Jövőkép: ReALIS-projekt

2006-ban úgy döntöttünk, hogy a két projektet egyesítjük, és felhasználva az eredményeiket, létrehozunk egy új, nagyobb méretű és hatékonyabban működő adatbázist. Az új projekt kiinduló célja továbbra is a totális lexikalizmus működőképességének igazolása elméletben és gyakorlatban. Az új rendszer struktúrájában (technológiájában) a LiLe adatbázisához hasonlít, a lényeges különbség a szemantikai reprezentációban lesz: míg a korábbi implementációk az LDRT-t tekintették a reprezentáció alapjának, az új projekt szemantikai komponensét egy még alaposabb interpretációs rendszer – a ReALIS (Alberti 2005) – alkotja.

A ReALIS-projekt elméleti célja tehát továbbra is a totális lexikalizmus legitimálása, viszont a hatékonyabb fejlesztőkörnyezetnek és az alaposabb szemantikai rendszernek köszönhetően bízunk abban, hogy gyakorlati célokat is meg tudunk majd valósítani. Ezek közül a legfontosabb egy olyan (megfelelően hatékony) elemző program létrehozása, amely első lépésként fókuszos mondatok kezelésére lenne képes (hozzájuk szintaktikai és szemantikai reprezentációt társítana); majd egyéb nyelvi jelenségeket tartalmazó mondatokat is elemezni tudna (a GeLexi-projekt e téren elért eredményeit felhasználva). Következő lépésként más nyelvek lexikai egységeivel is feltöltenénk az adatbázist (amelyet a nyelvfüggetlenséget szem előtt tartva terveztünk meg), és végső célként az intelligens gépi fordítást szeretnénk megvalósítani, először kis adatbázison, majd – ha úgy tűnik, hogy a mechanizmusok működnek – tesztelnénk a megközelítés hatékonyságát nagyobb szóállományon.

A munka első részfeladatát már teljesítettük: létrehoztuk az adatbázist, amely (reményeink szerint) kellően univerzális ahhoz, hogy bármely nyelv lexikai egységeit (és azok minden tulajdonságát) tartalmazni tudja. Jelenleg az elemző algoritmuson dolgozunk, amelyet implementálva tesztelhető lesz az adatbázis szerkezete, hogy ténylegesen minden tulajdonság tárolható-e benne. Ezzel párhuzamosan zajlik a különféle nyelvi jelenségek (elsőként a fókusz) kidolgozása.

Rendszerünk előnye és újszerűsége abban rejlik, hogy a lexikonban ugyanazon keretek között kezelhető a szintaxis körébe (is) tartozó számos tényező (predikátum-argumentum,

illetve régens-vonzat viszonyok, szabad határozók, szórend). A rangparaméterek elegánsan számot adnak az egy nyelven belüli szórendi variációkról (szón belül a morfémák sorrendjéről), valamint a nyelvek közötti különbségekről is. A domináns rangparaméterek használatával pedig a szórendet megvariáló, sokszor láthatatlan elemek (fókusz, progresszív) is kezelhetőek.

7 Összegzés

Az elmúlt néhány évben érdekes „kísérletet" végeztünk: megvizsgáltuk, mi történik, ha a nyelvleírásban egyre sikeresebb lexikalizmust a végsőkig fokozzuk; arra kerestük a választ, hogy egy „totálisan" lexikalista nyelvtan eredményes tud-e lenni elméletben, illetve gyakorlatban. A nyelvtan előnyei, hogy a hangsúlyt a szemantikára helyezi, kompozicionális a DRT-vel, egyszintű, homogén, és bármely nyelv leírására egyformán alkalmas. Annak igazolására, hogy a mechanizmusok működnek, készítettünk egy implementációt, amely magyar és angol nyelvű mondatokhoz képes többféle (közük szemantikai) reprezentációt társítani, és a két nyelv között a gépi fordítást is megvalósítja. Működő rendszerünk bizonyíték arra, hogy a kiinduló nyelvtan jól formalizált és egzakt, és hogy alapjain készíthetők számítógépes nyelvészeti alkalmazások. A következő lépés a megközelítés hatékonyságának a vizsgálata, amelyhez az adatbázis méretének jelentős növelése az egyik fontos feladat, és további nyelvi jelenségek kezelésének a részletes kidolgozása a másik. Ezeken fogunk dolgozni a jövőben, hogy bebizonyítsuk, hogy a totálisan lexikalista megközelítést érdemes alkalmazni a nyelvtechnológia területén is.

III. Tézisek

1. tézis (kiindulás): A nyelvtudomány fejlődését (a szintaxisközpontúságtól a lexikon és a szemantikai komponens egyre jelentősebb szerepéig) és a különféle lexikalista elméletek sikerét figyelembe véve van *létjogosultsága egy olyan nyelvtannak, amelyben csak lexikon van, és közvetlen módon képes eljutni egy szemantikai reprezentációig; érdemes tehát kipróbálni, hogy a GASG mennyire tudja felvenni a versenyt más elméletekkel elméletben és gyakorlatban.*

A dolgozatban megmutattam a lexikalista megközelítés előnyeit, röviden bemutattam a legfontosabb lexikalista elméleteket és jellemzőiket. Azt is megmutattam, hogy korábban is történtek törekvések a radikálisabb lexikalizmus felé (Karttunen (1986), bizonyos értelemben Schneider (2005)). A GASG egyik legfontosabb jellemzője a szemantika középpontba helyezése. Megmutattam, hogy sok elmélet tartja ezt egyre fontosabbnak (például MRS hozzárendelése a HPSG-hez, illetve újabban a többi lexikalista elmélethez is). Ha a fő cél a szemantikához való eljutás, előnyösebb egy olyan elmélet, mint a GASG, amelyben nem kell a szintaxis és a szemantika között bonyolult leképezéseket megfogalmazni.

2. tézis (elméleti cél): A sikeres implementáció igazolja, hogy a GASG egy egzakt formális rendszer, a megközelítés működőképes, vagyis pusztán jegyek unifikációjával eldönthető egy mondat grammatikalitása, és a helyes mondatokhoz szemantikai reprezentáció társítható.

Az implementáció előtt a GASG-ben mint elméleti keretben kevés nyelvi példa volt teljesen kidolgozva. Adott volt egy radikális (csak lexikai jegyekre hivatkozó) megközelítés, illetve egy olyan formalizmus, amelyből az implementáció során ki lehetett indulni. A legfontosabb különbség más lexikalista elméletekhez képest a rangparaméterek bevezetése, amelyek ugyanolyan jegyként vannak kezelve, mint például a szófaj vagy az eset, és amelyek (többek között) a szórend kialakításáért felelősek (ezek „váltják ki" a frázisstruktúra szabályokat). Az implementáció elsődleges célja annak bizonyítása volt, hogy az elmélet olyan részletességgel formalizálható, hogy működő nyelvtechnológiai alkalmazás építhető rá. Ehhez további példákat kellett kidolgozni, ki kellett találni, hogyan „kell" kezelni a különféle nyelvi jelenséget totálisan lexikalista módon. A kutatásban Alberti Gábor (a GASG megalkotója) is részt vett, így az elméleti tisztaság és a totálisan lexikalista megközelítés megőrzése biztosítva volt. Az implementáció több szinten is visszahatott tehát az elméletre. Egyrészt konkrétabb definíciók formájában, mint például a közvetlen megelőzésért felelős, rangparaméterekre hivatkozó *immprec*, valamint az elmélet nagyon pontos matematikai definíciója (Alberti et al. 2003) is az implementáció hatására íródott. Másrészt konkrét példák (különféle nyelvi jelenségek alapos kidolgozása) formájában, mint például a mellérendelés. Vagy (ami a legradikálisabb változás) a morfémaszintre való áttérést eredményezve, ami nem azt jelenti, hogy beépítettünk egy morfológiai komponenst, hanem – a még nagyobb fokú univerzalitás elérése érdekében – minden morféma külön lexikai egység, saját jegyekkel és követelésekkel (hasonló törekvés olvasható például Gambäck (2005)-ben). Az, hogy ezt a munkát el tudtuk végezni – létre tudtunk hozni egy működő implementációt az elméleti tisztaság megőrzése mellett – bizonyítja, hogy a GASG egy egzakt, formalizálható rendszer.

3. tézis (gyakorlati cél): Mivel a számítógépes nyelvészet területén még mindig sok a megoldatlan probléma (mint például a pontos, jó minőségű gépi fordítás), szükség van újabb és újabb módszerekre, megközelítésekre, amelyek segítségével a problémák közül néhány sikeresen megoldhatóvá válhat. A legígéretesebbek alkalmazások (intelligens

nyelvtechnológiai célokra) „igazi" nyelvészeti háttérrel dolgoznak, és lexikalista alapon működnek; érdemes tehát ezeket az új módszereket lexikalista keretben kidolgozni, illetve kipróbálni egy minden eddiginél szótárközpontúbb megközelítést nem csupán elméletben, hanem gyakorlatban is.

A dolgozatban bemutattam a számítógépes nyelvészeti kutatások jelenlegi állását, rámutattam, hogy intelligens célokra nyelvészeti alapú, mélyelemzést végző és szemantikai információt is hasznosító (sőt, szemantikai reprezentációt nyújtó) rendszerekre van szükség. Részletesebben bemutattam néhány lexikalista alapú alkalmazást, kitérve a még meglévő problémákra, és a jelenlegi kutatásokra. A magyarországi számítógépes nyelvészeti alkalmazásokról is szót ejtettem, felhívva a figyelmet arra, hogy sok rendszernek nem is célja alapos elemzést vagy igazán jó minőségű fordítást adni, olyan rendszerről pedig nincs tudomásom, amely minden információt visszaadó elemzést vagy fordítást kívánna nyújtani (retorikai viszonyok, fókusz). Így tehát mindenképpen érdemes további kutatásokat végezni, újabb módszereket kipróbálni.

4. tézis (eredmények, GeLexi-projekt): Amit elértünk: a GASG implementációjaként a GeLexi-projekt létrehozott egy olyan programot, amely egy beírt szósorról eldönti, hogy grammatikus mondatot alkot-e, és ha igen, különféle kimeneteket társít hozzá: morfofonológiai, szintaktikai és szemantikai. A lexikon magyar és angol nyelvű lexikai egységeket tartalmaz, és a két nyelv között a gépi fordítást is megvalósítja. Lehetséges tehát működő alkalmazást építeni totálisan lexikalista alapokon.

A dolgozatban részletesen bemutattam a GeLexi-projekt által készített Prolog-implementációt. Különféle nyelvi jelenségeket tartalmazó mondatok elemzését és fordítását ismertettem, mint például műveltetés, zéró névmások, igekötő, vonzatos melléknév és mellérendelés. A többértelműség kezelésére is hoztam példát, illetve részletesebben ismertettem, hogyan lehet a morfofonológiát totálisan lexikalista keretben kezelni. Mivel a cél elsőként a megközelítés működőképességének a bizonyítása volt, az adatbázis csupán néhányszáz lexikai egységet (magyar és angol nyelvű morfémákat) tartalmaz.

Részletesebben a különféle részfeladatokról:

a) morfofonológia

Kezdetben a program (akárcsak az elmélet) lexikai egységei (ragozott) szavak voltak, később azonban áttértünk morfémaszintre, aminek két oka volt. Az egyik a hatékonyság, a másik pedig annak az elméleti állításnak a bizonyítása, hogy lehetséges a morfológiát is totálisan lexikalista módon kezelni.

Az elméleti háttért Alberti Gábor dolgozta ki. A megközelítés lényege, hogy minden morfémának lehet követelménye és hozzájárulása a szemantikához (proto-DRS-e), ami nagyobb fokú univerzalitást tesz lehetővé. Például a műveltetés egyes nyelvek esetében önálló szó, máshol pedig affixum. Ez a különbség ebben a megközelítésben eltűnik, így a különféle nyelvi jelenségek egységesebb módon kezelhetők.

Az implementációs munkát alapvetően én végeztem: az elmélet alapján kidolgoztam az egyes lexikai egységek pontos leírásait – hogy melyik morfémának mi legyen a sajátszava, és milyen (morfo)fonológiai jegyei és elvárásai legyenek –, majd mindezt számítógépre vittem. Az eredményekről a düsseldorfi ICSH-n tartottam előadást 2002-ben, a hozzá tartozó cikk négyünk neve alatt futott (Alberti–Balogh–Kleiber–Viszket 2005). Ezután a programot (szintaxist, szemantikát) át kellett írni ez alapján, amit közösen végeztük Balogh Katával.

b) szintaxis
A második kimenet a függőségi viszonyokat tartalmazza (régensek és vonzatok). Ezt közösen írtuk Balogh Katával (Alberti Gábor javaslatai alapján). A program ellenőrzi a különféle szintaktikai jegyeket (megfelelő eset, egyeztetés), illetve a szórendet is a közvetlen megelőzést kódoló rangparaméterek alapján, amelyek ugyanolyan jegyek, mint például a szófaj vagy az eset, ami által egy nagyobb fokú homogenitás érhető el. Ha minden tulajdonság és elvárás unifikálódni tud, a program kiírja, hogy melyik morféma melyik másikkal milyen szintaktikai viszonyt létesít (alany, tárgy, szabad bővítmény stb.).

c) szemantika
A harmadik kimenet a szemantikai reprezentáció – egy DRS-szerű ábrázolás. Az elméleti háttér az LDRT (Alberti 2000), ami a hagyományos DRT kiterjesztése. A gyakorlati megvalósítás alapvetően Balogh Kata munkája (Alberti Gábor javaslatai alapján), később én többször átdolgoztam, amikor változtattuk vagy bővítettük a programot. A program kiírja a mondat által bevezetett referenseket és a hozzájuk tartozó kondíciósorokat, valamint egy rendezést a világok között (mint a DRS-nél a dobozstruktúra). A reprezentáció nyelve angol, így a program a géppel segített fordítás területén is hasznosítható. A szemantikával kapcsolatos megközelítésünket és az elért eredményeinket Alberti–Balogh–Kleiber–Viszket (2003) ismerteti először.

d) fordítás
A fordítás alapötlete, hogy ha elkészítjük a forrásnyelvi mondat elemzését, abból generálni tudunk olyan célnyelvi mondatot, amely ugyanazokat az információkat tartalmazza. Nincs szükség tehát külön „szabályrendszerre" minden nyelvpár (és irány) esetében, csupán a különféle nyelvek lexikai egységeit kell rögzíteni minden tulajdonságukkal együtt, és bármely két nyelv között lehetséges lesz a fordítás.

Még egy reprezentációs szintet létrehoztunk annak érdekében, hogy fordításkor pontosan azokat az információkat vigyük át, amikre szükség van. Ez lett a kopredikációs hálózat, amely azt rögzíti, hogy mely elemek tesznek ugyanarról állítást. Az elméleti háttér itt is Alberti Gábor munkája, a megvalósítást Balogh Katával együtt végeztük.

Hogy kipróbálhassuk a megközelítés működőképességét, angol nyelvű lexikai egységeket is fel kellett venni az adatbázisba, ezt alapvetően én végeztem. A dolgozatban is megmutattam, hogy egy adott mondat kopredikációs hálózata csak a morfémák számozásában tér el a két nyelv esetében.

A fordítás konkrét mechanizmusát Alberti Gáborral együtt dolgoztuk ki, és én valósítottam meg. A máltai EAMT konferencián én adtam elő a fordítás totálisan lexikalista megközelítését és az elért eredményeinket (a hozzá kapcsolódó cikk: Alberti–Kleiber 2004).

5. tézis (eredmények, LiLe-projekt): A GASG-t modernebb és hatékonyabb környezetben is implementáltuk, létrehoztunk egy relációs adatbázist és egy Delphi nyelvű elemzőprogramot, amelyek segítségével magyar szavak morfofonológiai elemzését tudtuk elvégezni (néhány speciális lehetőséggel, mint a hibás variációk kiíratása, feltüntetve a hiba okát is, mellyel az oktatást kívántuk támogatni).

A GeLexi-projekt által készített rendszer hatékonyságát nem teszteltük, úgy gondoltuk, miután a mechanizmusok működőképességét bizonyítottuk, modernebb fejlesztői környezetbe helyezve lenne érdemes kiértékelést végezni. (Részben) erre a célra alakult meg a LiLe-projekt, amely a megvalósításban a morfofonológiai komponensig jutott (így a hatékonyság tesztelése sem történt meg). Jelentősége a kiinduló célok megvalósítása szempontjából, hogy kidolgozott egy modernebb struktúrát nem csupán a morfofonológia, hanem a többi nyelvi szint elemzéséhez is, amelyet a jelenlegi munkánkban is hasznosítunk (ReALIS-projekt).

IV. Hivatkozások

Alberti, Gábor (1999): GASG: The Grammar of Total Lexicalism. In: *Working Papers in the Theory of Grammar* 6/1, Theoretical Linguistics Programme, Budapest University and Research Institute for Linguistics, Hungarian Academy of Sciences.

Alberti, Gábor (2000): Lifelong Discourse Representation Structures. In: *Gothenburg Papers in Computational Linguistics* 00–5, Sweden, 13–20.

Alberti Gábor (2005): *ReALIS*. Akadémiai doktori értekezés.

Bender, Emily M., Dan Flickinger, and Stephan Oepen (2002): The Grammar Matrix: An Open-Source Starter-Kit for the Rapid Development of Cross-Linguistically Consistent Broad-Coverage Precision Grammars. In: *Proceedings of COLING 2002 Workshop on Grammar Engineering and Evaluation*, Taipei.

Bond, Francis, Stephan Oepen, Melanie Siegel, Ann Copestake, Dan Flickinger (2005): Open source machine translation with DELPH-IN. In: *Proceedings of the Open-Source Machine Translation Workshop at the 10th Machine Translation Summit*, Phuket, Thailand, 15–22.

Butt, Miriam, Helge Dyvik, Tracy Holloway King, Hiroshi Masuichi, and Christian Rohrer (2002): The Parallel Grammar project. In: *Proceedings of COLING 2002 Workshop on Grammar Engineering and Evaluation*, Taipei.

Copestake, Ann, Dan Flickinger, Ivan Sag, Carl Pollard (2005): Minimal Recursion Semantics: An introduction. In: *Research in Language and Computation* 3(2–3), 281–332.

Frank, Anette (2003): Projecting LFG F-structures from Chunks or (Non-)Configurationality from a different Viewpoint. In: Miriam Butt, Tracy Holloway King (eds.): *The Proceedings of the LFG'03 Conference*, University at Albany, State University of New York, CSLI Publications, 217–237.

Forst, Martin, Jonas Kuhn, Christian Rohrer (2005): Corpus-Based Learning of OT Constraint Rankings for Large-Scale LFG Grammars. In: Miriam Butt, Tracy Holloway King (eds.): *Proceedings of the LFG'05 Conference*, University of Bergen, CSLI Publications, 154–165.

Gambäck, Björn (2005): Semantic Morphology. In: *Inquiries into Words, Constraints and Contexts: Festschrift in the Honour of Kimmo Koskenniemi on his 60th Birthday,* CSLI Studies in Computational Linguistics, CSLI Publications, Stanford, California, 204–213.

Joshi, Aravind K. (2003): Starting with complex primitives pays off. In: Alexander Gelbukh (ed.): *Proceedings of CICLing2003*, Springer-Verlag. 1–10.

Kálmán László, Balázs László, Erdélyi Szabó Miklós (2003): Tudásalapú természetesnyelv-feldolgozás. In: Alexin Zoltán, Csendes Dóra (szerk): *Magyar Számítógépes Nyelvészeti Konferencia MSZNY 2003*, Szegedi Tudományegyetem, Egyetemi nyomda, Szeged, 109–114.

Kálmán László, Trón Viktor, Varasdi Károly (szerk.) (2002): *Lexikalista elméletek a nyelvészetben*. Tinta Könyvkiadó, Budapest.

Karttunen, Lauri (1986): *Radical Lexicalism,* Report No. CSLI 86–68, Stanford.

Komlósy András (2001): *A lexikai-funkcionális grammatika mondattanának alapfogalmai*. Tinta Könyvkiadó, Budapest.

Mitkov, Ruslan (szerk.) (2003): *The Oxford Handbook of Computational Linguistics.* Oxford University Press, New York.

Pinker, Stephen (1999): *Words and Rules: The Ingredients of Language*. New York, HarperCollins.

Prószéky Gábor (2005): A világháló nyelvi vizsgálata. In: Alexin Zoltán, Csendes Dóra (szerk): *III. Magyar Számítógépes Nyelvészeti Konferencia MSZNY 2005*, Juhász Nyomda, Szeged, 3–12.

Prószéky, Gábor, László Tihanyi, Gábor Ugray (2004): Moose: a robust high-performance parser and generator. In: John Hutchins (ed.): *Broadening Horizons of Machine Translation and its Applications. Proceedings of the Ninth EAMT workshop*, Foundation for International Studies, University of Malta, Valletta, 138–142.

Schneider, Gerold (2005): A Broad-Coverage, Representationally Minimal LFG Parser: Chunks and F-structures are Sufficient. In: Miriam Butt and Tracy Holloway King (eds): *Proceedings of the LFG'05 Conference*, University of Bergen, CSLI Publications, 388–407.

Trón Viktor (2001): *Fejközpontú frázisstruktúra-nyelvtan*. Tinta Könyvkiadó, Budapest.

van Eijck, Jan, Hans Kamp (1997): Representing Discourse in Context. In: Johan van Benthem, Alice ter Meulen (eds.): *Handbook of Logic and Language,* Elsevier, Amsterdam, MIT Press, Cambridge, Mass., 179–237.

V. A témához kapcsolódó saját publikációk

Alberti, Gábor, Kata Balogh, Judit Kleiber (2002a): GeLexi Project: Prolog Implementation of a Totally Lexicalist Grammar. In: Dick de Jongh, Marie Nilsenova, Henk Zeevat (eds.): *Proceedings of the Third and Fourth Tbilisi Symposium on Language, Logic and Computation*, ILLC, Amsterdam, and Univ. of Tbilisi.

Alberti Gábor, Balogh Kata, Kleiber Judit, Viszket Anita (2002b): A totális lexikalizmus elve és a GASG nyelvtan-modell. In: Maleczki Márta (szerk.): *A mai magyar nyelv leírásának újabb módszerei* V., SZTE, Szeged, 193–218.

Alberti, Gábor, Kata Balogh, Judit Kleiber, Anita Viszket (2003): Total Lexicalism and GASGrammars: A Direct Way to Semantics. In: Alexander Gelbukh (eds.): *Proceedings of CICLing2003*, Springer-Verlag, 37–48.

Alberti, Gábor, Kata Balogh, Judit Kleiber, Anita Viszket (2005): Towards a Totally Lexicalist Morphology. In: István Kenesei, Christopher Piñón (eds.): *Approaches to Hungarian* 9, 9–33.

Alberti Gábor, Balogh Kata, Kleiber Judit, Viszket Anita (2007a): A fordítás totálisan lexikalista megközelítése. In: Fóris Ágota és Tóth Szergej (szerk.): *Terminologia et Corpora – Supplementum: Ezerarcú lexikon*, Szombathely, 143–152.

Alberti Gábor, Dóla Mónika, Kántor Gyöngyi, Kleiber Judit, Ohnmacht Magdolna (2007b): ReALIS: a „reális" interpretációs rendszer. In: Alberti Gábor, Fóris Ágota (szerk.): *A mai magyar formális nyelvtudomány műhelyei*, Nemzeti Tankönyvkiadó, Budapest, 139–156.

Alberti Gábor, Kleiber Judit (2001): Világok között az Életfogytiglani DRS-ben. In: Kabán Annamária (szerk.): *Funkcionális mondatperspektíva és szövegszerkesztési stratégia*, Miskolci Egyetemi Kiadó, 35–46.

Alberti, Gábor, Judit Kleiber (2003): Extraction of Discourse-Semantic Information from Hungarian Sentences by means of a Totally Lexicalist Grammar. In: Elena Paskaleva, Galia Angelova, Hamish Cunningham, Kalina Bontcheva (eds.): *Information Extraction for Slavonic and Other Central and Eastern European Languages*, Borovets, Bulgaria, 63–69.

Alberti, Gábor, Judit Kleiber (2004): The GeLexi MT Project. In: John Hutchins (eds.): *Proceedings of EAMT 2004 Workshop*, Valletta: Univ. of Malta, 1–10.

Alberti Gábor, Kleiber Judit, Ohnmacht Magdolna, Szilágyi Éva, Anne Tamm, Viszket Anita (2006): ReALIS projekt: a szóképzés általánosítása a számítógépes fordításban. In: Alexin Zoltán, Csendes Dóra (szerk.): *IV. Magyar Számítógépes Nyelvészeti Konferencia MSZNY 2006*, Juhász Nyomda, Szeged, 41–51.

Alberti Gábor, Kleiber Judit, Viszket Anita (2003): GeLexi Projekt: GEneratív LEXIkonon alapuló mondatelemzés. In: Csendes Dóra, Alexin Zoltán (szerk.): *Magyar Számítógépes Nyelvészeti Konferencia MSZNY 2003*, Szegedi Tudományegyetem, Egyetemi Nyomda, Szeged, 79–85.

Alberti, Gábor, Judit Kleiber, Anita Viszket (2004a): GeLexi project: Sentence Parsing Based on a GEnerative LEXIcon. In: *Acta Cybernetica* 16, 587–600.

Alberti Gábor, Kleiber Judit, Viszket Anita (2004b): GeLexi projekt: Fordítás totálisan lexikalista alapokon. In: Alexin Zoltán, Csendes Dóra (szerk): *II. Magyar Számítógépes Nyelvészeti Konferencia MSZNY 2004*, Juhász Nyomda, Szeged, 73–80.

Balogh, Kata, Judit Kleiber (2003): Computational Benefits of a Totally Lexicalist Grammar. In: Pavel Mautner, Vaclav Matousek (eds.): *Text, Speech and Dialogue, Proceedings of TSD2003*, Springer-Verlag, Berlin Heidelberg New York, 114–119.

Balogh, Kata, Judit Kleiber (2003): A Morphology Driven Parser for Hungarian. In: Rusudan Asatiani, Kata Balogh, George Chikoidze, Paul Dekker, Dick de Jongh (eds.): *Proceedings of the Fifth Tbilisi Symposium on Language, Logic and Computation*, ILLC, Amsterdam, and Univ. of Tbilisi.

Bódis Zoltán, Kleiber Judit, Szilágyi Éva, Viszket Anita (2003): *Nyelvészeti lexikon – oktatási és kutatási adatbázis fejlesztése (LILE projekt)*. Előadás a 'Multimédia az oktatásban' c. konferencián. Elérhető: http://lingua.btk.pte.hu/lile.asp

Bódis Zoltán, Kleiber Judit, Szilágyi Éva, Viszket Anita (2004): LiLe projekt: Adatbázis mint „dinamikus korpusz". In: Alexin Zoltán, Csendes Dóra (szerk): *II. Magyar Számítógépes Nyelvészeti Konferencia MSZNY 2004*, Juhász Nyomda, Szeged, 11–18.

Kleiber, Judit (2005): Across world(let)s in a representationist interpretation system. In: Judit Gervain (ed.): *Proceedings of the Tenth ESSLLI Student Session*, 8–19 August 2005, Edinburgh, UK. 112–121.

Kleiber Judit (2006): Szótár totálisan lexikalista alapokon. In: Kassai Ilona (szerk.): *Szakszó, szaknyelv, szakmai kommunikáció*, Nyelvészeti Doktorandusz Füzetek 3, 18–31.

Kleiber, Judit (2007a): Total Lexicalism in Language Technology. In: Ville Nurmi, Dmitry Sustretov (eds.): *Proceedings of the Twelfth ESSLLI Student Session*, 6–17 August 2007, Dublin, Ireland. 149–160.

Kleiber Judit (2007b): Számítógépes nyelvészet Pécsen. In: Alberti Gábor, Fóris Ágota (szerk.): *A mai magyar formális nyelvtudomány műhelyei*, Nemzeti Tankönyvkiadó, Budapest, 170–188.

Szilágyi Éva, Kleiber Judit, Alberti Gábor (2007): A totálisan lexikalista szintaxis rangja(i). In: Tanács Attila, Csendes Dóra (szerk.): *V. Magyar Számítógépes Nyelvészeti Konferencia MSZNY 2007*, Juhász Nyomda, Szeged, 284–287.

VI. Függelék: Saját hozzájárulásom a közös munkához

A dolgozatban leírt eredmények mindig egy csapat munkájához köthetők, sok helyen lehetetlen szétválasztani, hogy egy adott eredmény kinek köszönhető. Van azonban néhány olyan munka, amelyről pontosan meg lehet mondani, hogy ki végezte. A dolgozatban nem tartottam fontosnak ezeket ismertetni, ezért itt teszem meg (illetve részben a tézisekben már említettem). Először leírom, egy-egy fontos eredmény alapvetően kinek köszönhető, majd összefoglalom, hogy mi az én hozzájárulásom a közös munkához.

1. Az elmélet (GASG) Alberti Gábor nevéhez köthető, de több ponton visszahatott rá az implementáció (ami közös munka). A legjelentősebb a morfémaszintre való áttérés, de az egyik legfontosabb definíció – a közvetlen megelőzési viszonyokat ellenőrző *immprec* – is a programból került az elméletbe. Illetve a részletek kidolgozása is az implementációnak köszönhető (vonzatos melléknév, mellérendelés stb.).
2. Az ötlet, hogy készüljön egy implementáció, szintén Alberti Gáboré, a Prolog programnyelvet is ő javasolta, mert a „motorja" ugyanúgy az unifikáció, mint az elméleti modellé (a totálisan lexikalista GASG-é).
3. Kezdetben a programozási munkát teljesen együtt végeztük Balogh Katával, illetve Alberti Gábor is részt vett időnként az implementálásban (az új jelenségek kezelésének ötletét gyakran majdnem hogy programsorokban fogalmazta).
4. Többé-kevésbé önálló munkának számít Balogh Katának a szemantikai elemzés és reprezentáció, nekem pedig a morfofonológia (lexikai egységek kidolgozása, implementáció).
5. Miután Balogh Kata kivált a csoportból (2003 nyarán), egyedül folytattam az implementálást, így az ezután készült programverziók önálló munkámnak számítanak (mellérendelés és fordítás). Az elméleti kiindulópont természetesen ezek esetében is Alberti Gáboré, az implementáláshoz szükséges részleteket közösen dolgoztuk ki, új tagunknak, Viszket Anitának pedig alapvetően a kritikusi szerep jutott.
6. A LiLe-projekt munkája teljesen közös, szétválaszthatatlan (Viszket Anitáé a legnagyobb érdem). Minden alkalommal, amikor az adatbázison dolgoztunk, minden tag jelen volt, továbbá a cikkeket is úgy írtuk, hogy felosztottuk, és mindenki ugyanannyi oldalt írt.
7. A ReALIS adatbázisát Szilágyi Évával közösen készítettük (szétválaszthatatlan), időnként Alberti Gáborhoz fordultunk elméleti tisztázásért. A program (algoritmus) önálló munkám.

Mely pontjai köthetők kimondottan hozzám:

a) lexikalista beágyazás (elméletek és alkalmazások bemutatása) – a dolgozat maga (a „krónikási" szerep)
b) implementáció készítésében való részvétel (átlag 50%)
c) önálló implementációs munka: morfofonológia, mellérendelés, fordítás.