Doktori (Ph.D.) értekezés tézisei

A magyar nyelv automatikus szintaktikai elemzése szabályalapú gépi tanulási technikák alkalmazásával

Hócza András

Témavezető: *Gyimóthy Tibor, PhD*

Szegedi Tudományegyetem TTIK, Matematika- és Számítástudományok Doktori Iskola

Szeged, 2008

Bevezetés

Az összefoglaló ismerteti a „*A magyar nyelv automatikus szintaktikai elemzése szabályalapú gépi tanulási technikák alkalmazásával*” című Ph.D. disszertáció eredményeit. A disszertáció témája a szintaktikai elemzés, melynek a gyakorlati megvalósítása és alkalmazása magyar nyelvre történt. A szerző módszereiben szabályalapú gépi tanulási technikákat alkalmazott, melyek segítségével egy elemzett korpuszból kinyerhető információk felhasználásával szintaktikai elemzésre alkalmazható modell építhető. A szabályalapú reprezentáció ember számára olvasható módon tárolja a megszerzett ismereteket, így lehetőséget biztosít a tudásbázis karbantartására és szakértői tudással történő kiegészítésre.

A természetes nyelvek szintaktikai elemzése

A természetes nyelvek jelenségeinek ábrázolása kihívást jelent a számítógépes nyelvészet számára, ezen belül a magyar nyelv a szabad szórend és a ragozott szóalakok nagy száma miatt a nehezebben elemezhető nyelvek közé sorolható. A nyelvtani formalizmusok olyan metanyelvek, amelyek definiálják azt a rendszert, amellyel egy természetes nyelv szabályai leírhatók. Ezekkel szemben a következő követelményeket támaszthatjuk:

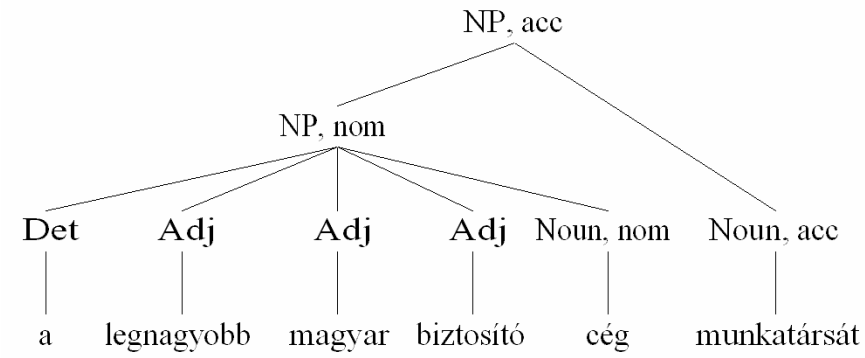
- ***Nyelvészeti alkalmasság***: annak mértéke, hogy az adott metanyelven mennyire lehet egyes nyelvi jelenségeket a nyelvészek által alkalmazott elveknek megfelelően kifejezni.
- ***Számítási hatékonyság***: annak mértéke, hogy az adott nyelvtani formalizmus milyen hatékonyan valósítható meg számítógépen.

A generatív nyelvtanok alkalmazásai megjelenésük idején ígéretes lehetőségnek tűntek, mivel ezeket hatékony algoritmusokkal lehet elemezni, különösen a reguláris és a környezetfüggő nyelvtanok esetén. Azonban hamarosan megjelentek cáfolatok, ellenpéldák, melyek azt mutatták, hogy ezek a nyelvosztályok nem alkalmasak a természetes nyelvek bizonyos jelenségeinek ábrázolása. Ilyen ellenpélda az önbeágyazás, melynek leírására nem alkalmas a reguláris nyelvtan, a keresztező függőségek leírását pedig a környezetfüggő nyelvtanokkal nem lehet megoldani.

Napjainkra a generatív megközelítés háttérbe szorult, ezeket felváltották olyan nyelvelméletek és formalizmusok, melyekben a nyelvi jelenségek minél pontosabb leírása került előtérbe a nyelv generálása helyett. Az egyeztetés és alkategórizálás,

például azért jelent problémát, mert környezetfüggetlen nyelvtan alkalmazása esetén csak nagyon sok szabály bevezetésével lehetne leírni ezt a jelenséget, ez a megoldás viszont a nyelvtan méretét a többszörösére növelné. Szintén ilyen jellegű problémát jelent a szabad szórend kezelése. A függőségek ábrázolása, különösen a távoli függőségeké nem oldható meg környezetfüggetlen nyelvtanokkal, mert ezek szabályai csak összefüggő szócsoportokra alkalmazhatóak. A szabályok alkalmazásának statisztikai előfordulások alapján becsült valószínűségét is figyelembe kell vennünk, ha olyan modellt szeretnénk készíteni, ami alapján választani tudunk a többértelműségek miatt kialakult elemzési erdőből. Végezetül a lexikális és strukturális függőségek figyelmen kívül hagyása olyan elemzési szerkezeteket eredményezhet a gépi elemzésben, melyeket az annotált korpusz nem tartalmaz, mert ezeket a szöveg értelmezése alapján kizárhatjuk.

A szerző által bevezetett faminta formalizmus többszintű részfákat ismer fel a leveleire adott reguláris kifejezésekkel leírt minták segítségével. Tegyük fel, hogy adott egy többszintű fa (1. ábra), továbbá a szavakhoz illetve szócsoportokhoz hozzá van rendelve azok esete is (nom - alany, acc - tárgy).



1. Ábra. Egy többszintű főnévi csoport

A fa által lefedett szócsoportokon mindenféle transzformációkat hajthatunk végre, elhagyhatunk, beszúrhatunk, átrendezhetünk és kicserélhetünk szavakat. Így az eredeti szócsoporthoz nagyon hasonló szócsoportokat kapunk, valamint látjuk azt, hogy hol vannak azok a pontok ahol variálhatjuk a leveleket, anélkül, hogy a magasabb szintek szerkezetén változtatni kellene. További hasonló esetek lehetnek például:

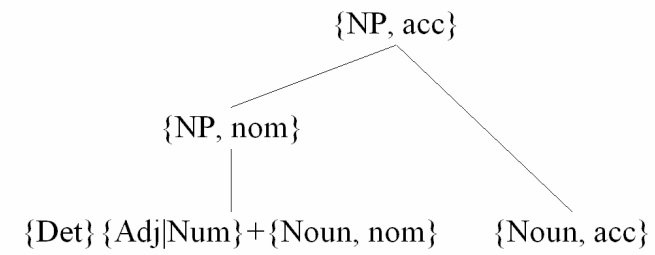
$a_{\{Det\}}$ legnagyobb$_{\{Adj\}}$ biztosító$_{\{Adj\}}$ cég$_{\{Noun,nom\}}$ munkatársát$_{\{Noun,acc\}}$

$a_{\{Det\}}$ $2_{\{Num\}}$ legnagyobb$_{\{Adj\}}$ biztosító$_{\{Adj\}}$ cég$_{\{Noun,nom\}}$ munkatársát$_{\{Noun,acc\}}$

$a_{\{Det\}}$ $2_{\{Num\}}$ cég$_{\{Noun,nom\}}$ munkatársát$_{\{Noun,acc\}}$

$az_{\{Det\}}$ első$_{\{Num\}}$ $2_{\{Num\}}$ cég$_{\{Noun,nom\}}$ munkatársát$_{\{Noun,acc\}}$

Az előző pontokban felsorolt eseteket lefedhetjük egyetlen famintával, ami még ráadásul általánosít is, mert lefedi az előzőekben fel nem sorolt eseteket (2. ábra).



2. Ábra. A hasonló szerkezeteket lefedő faminta

Ha szintaktikai elemzésre többszintű szerkezeteket alkalmazunk a modell várhatóan több elemet (szabályt) fog tartalmazni egy ugyanolyan korpuszon felkészített környezetfüggetlen nyelvtanhoz képest. A faminta formalizmus ezt a növekedést azzal kompenzálja, hogy a faminták a levelek leírása révén, képesek egymáshoz hasonló szerkezetek csoportját összefoglalni egyetlen mintában. A famintákban szereplő leírás rugalmassága lehetővé teszi más formalizmusok nyújtotta technikák alkalmazását, amivel kezelni lehet a strukturális függőségeken túl más problémás nyelvi jelenségeket is. A szövegek szintaktikai elemzését a ***chart parser*** algoritmus ([Kaplan73], [Kay86]) famintákra adaptált változata valósítja meg.

Gépi tanulási technikák alkalmazása nyelvtani modellek készítésére

A gépi tanulási módszerek egyik fontos alkalmazási területe a természetes nyelvi problémák, különösen akkor, ha erre a célra rendelkezésre áll egy annotált korpusz, melyből példákat gyűjthetünk egy adott jelenségre. A példák halmaza olyan (x_i, y_i) értékpárokból áll, melyekben az x_i értékek valamilyen objektum vagy esemény leírására szolgálnak, az y_i értékek pedig a következtetést adják meg. Diszkrét y_i értékek esetén ***osztályozásról*** beszélhetünk. Azt az esetet ***felügyelt tanulásnak*** nevezzük, amikor az (x_i, y_i) értékpárok halmaza ismert (például az annotált korpuszból kigyűjthető) és a tanulóprogram feladata egy olyan f függvény megkeresése, melyre $f(x_i) = y_i$ teljesül. Ebben az esetben azt is feltételezzük, hogy f függvény alkalmas lesz előre nem látott x értékek esetén is az y értékek helyes meghatározására. Ezt az elvet ***induktív tanulásnak*** nevezzük. Amikor a cél egy logikai értékű osztályozás tanulása, ezt ***fogalom tanulásának*** hívjuk, ebben az esetben pozitív és negatív példáink vannak attól függően, hogy igaz vagy hamis érték van hozzájuk rendelve.

A szerző által kidolgozott ***RGLearn*** mintatanuló algoritmus bemenete az annotált korpuszból kigyűjtött mintákból képzett pozitív és negatív példák, az alapján, hogy

helyes, vagy hibás fedésről van-e szó. A kimenet egy olyan általánosított mintahalmaz, melynek együttes pontossága maximális, azaz a lehető legtöbb pozitív és a lehető legkevesebb negatív példát fedi le. Ez az algoritmus alkalmazva volt szófaji egyértelműsítés szabályalapú modelljének [Kuba04], valamint szintaktikai elemzésre használt famintáknak ([Hócza04a], [Hócza06a]) a tanulására is.

A RGLearn algoritmus a pozitív példákból különböző mértékű általánosítással kapható mintákhoz egy pontszámot rendel, hogy azokat rangsorolni lehessen az annotált korpuszon mért statisztika alapján. Ez a pontszám egy adott szempont szerinti mértéken alapul, több szempont esetén pedig vesszük a mértékek lineáris kombinációját, például:

$$score = \lambda_1 * (pos - neg) \ / pos + \lambda_2 * pos \ / \ (pos + neg) \qquad (1)$$

ahol a *pos* a lefedett pozitív példák száma, *neg* a lefedett negatív példák száma, valamint $\lambda_1 + \lambda_2 = 1$. Különböző λ_i értékekkel az algoritmus különböző szempontoknak megfelelő mintahalmazt állít elő, így ezek olyan paraméterei lehetnek az algoritmusnak, melyet az elemzés pontossága szerint lehet optimalizálni.

A gépi tanulási módszereknek további alkalmazási lehetőségei is vannak a szintaktikai elemzésre alkalmazható modellek építése során. A szófaji egyértelműsítésnél használt ***címkéző algoritmus (tagger)*** szócsoportok határainak jóslására is alkalmazható. A feladat például NP határok jóslására esetén úgy fogalmazható meg, hogy egy adott szópozícióhoz annak környezete alapján rendeljünk hozzá a következő 5 címke valamelyikét: NP eleje (B), NP belső szava (I), NP vége (E), egy tagú NP (BE) vagy NP-n kívül esik (O). Ez lényegében egy HMM-el megoldható címkézési feladat [Charniak93], vagy felügyelt tanulással (például C4.5 [Quinlan93]) megoldható osztályozási feladat, vagy több módszer kombinációja optimalizált súlyok szerint történő szavazással, általában ez utóbbi módszerrel lehet elérni a legnagyobb pontosságot. A szócsoportok határainak jóslási eredményét felhasználhatjuk a ***felszíni elemzés* (*Shallow Parsing*)** során a mondatok szócsoportokra való szegmentálására vagy az alapvető szócsoportok (például ***base-NP***, ***top-NP***) kijelölésére.

A minták halmazára készíthetünk egy valószínűségi modellt, melyet annotált korpusz esetén a ***relatív gyakoriságok*** alapján számíthatunk ki, amiről megmutatható (részletes bizonyítás: [Prescher03]), hogy ez a ***maximum likelihood*** becslést adja, azaz korpusz valószínűsége az így becsült valószínűségekkel kiértékelve lesz maximális. Annotált korpusz hiányában a valószínűségeket az ***Inside-Outside algoritmus*** [Baker79] segítségével közelíthetjük.

A modell mintáinak az összetétele változtatható ha a komplex modellkészítési folyamatot paraméterezhetővé tesszük és az elemzési pontosságra maximalizáljuk. Egy erre alkalmazható optimalizáló algoritmus a ***szimulált hűtés (Simulated Annealing)***

[Aarts89]. Különféle osztályozó módszerek kombinációjával feljavíthatjuk az egyedi módszerek eredményeit. További javulás érhető el az eredményekben, ha a módszerekhez súlyokat rendelünk és ezeket a példák egy részén a kiértékelés alapján optimalizáljuk.

Faminta alapú komplex szintaktikai elemző módszer

A szerző az automatikus szintaktikai elemzés megvalósítására kidolgozott egy komplex szintaktikai elemző módszert, mely a feladat részproblémáit összefoglalta egy összefüggő, paraméterezhető rendszerbe.

A modell építése a kiinduló faminta halmaz korpuszból való kigyűjtésével indul, mely faalak típusok felhasználásával történik. Jellemző faalak típusok például a ***beágyazás*** és a ***füzér*** melyek a korpuszból kigyűjtött részfák tulajdonságaira adnak meg kritériumokat. A faalak típusok előírásai összefüggő rendszert alkotva vezérlik a szintaktikai szerkezetek kigyűjtését a korpuszból, alkalmazásukkal tetszőleges elemzési fa lebontható. Egy példamondat feldolgozása a 3. ábrán látható.

Példamondat:

$[_{CP}\ [_{NP}\ [_{NP}\ \text{Mihály}_{Noun}\ ]\ \text{és}_{Conj}\ [_{NP}\ \text{az}_{Det}\ \text{ügyvéd}_{Noun}\ ]\ ]\ [_{VP}\ \text{felkereste}_{Verb}\ ]$
$[_{NP}\ \text{a}_{Det}\ [_{ADJP}\ \text{budapesti}_{Adj}\ ]\ \text{egyesület}_{Noun}\ ]\ \text{elnökét}_{Noun}\ ]\ .{}_{Punct}\ ]$

Kinyerhető minták:

füzér: $[_{NP}\ [_{NP}\ \text{Mihály}_{Noun}\ ]\ \text{és}_{Conj}\ [_{NP}\ \text{az}_{Det}\ \text{ügyvéd}_{Noun}\ ]\ ]$

beágyazás: $[_{VP}\ \text{felkereste}_{Verb}\ ]$

beágyazás: $[_{NP}\ [_{NP}\ \text{a}_{Det}\ [_{ADJP}\ \text{budapesti}_{Adj}\ ]\ \text{egyesület}_{Noun}\ ]\ \text{elnökét}_{Noun}\ ]$

A mondat leírása a kinyert minták behelyettesítése után:

$[_{CP}\ \text{NP VP NP}\ .{}_{Punct}\ ]$

Kinyerhető minták:

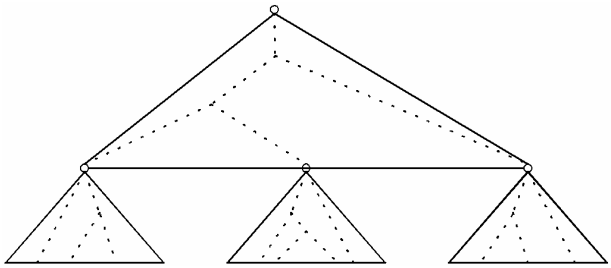
beágyazás: $[_{CP}\ \text{NP VP NP}\ .{}_{Punct}\ ]$

3. Ábra. Faalak-típusokkal végzett faminta-gyűjtés egy annotált példamondatból.

Mivel a teljes korpuszon nagyon sok különböző kigyűjtött részfa előállhat, ennek a nagy adattömegnek az együttes kezelése komoly technikai problémát jelentene. Ezért a részfákat csoportosítjuk a legáltalánosabb alakjuk alapján és a famintákat tanulását csoportonként végezzük el az RGLearn algoritmussal.

Az így kapott faminta halmazt a ***chart parser*** ([Kaplan73], [Kay86]) módosított változatával alkalmazzuk a szintaktikai elemzés során. Ez csak néhány kisebb

változtatást jelent az eredeti algoritmushoz képest. Az elemzés bottom-up stratégia szerint történik. A derivációs fában a faminták általában egy nagyobb összefüggő szintaktikai szerkezetet tartalmaznak, ezért ezek belső csúcsaira nem illesztünk famintát (4. ábra). Más szempontból ennek az elvnek köszönhetően az elemzési idő is csökkenthető.



4. Ábra. A famintákkal felismert részfák egymáshoz kapcsolódása

A szintaktikai elemzést a PARSEVAL metrikákkal [Black91] értékeljük ki úgy, hogy nem annak hibáját, hanem a jóságát fejezzük ki, azaz, hogy a felismert szócsoportok közül mennyi a helyes, ez a ***pontosság (precision)***, illetve, hogy a referencia elemzésben található (helyes) szócsoportok közül mennyit talált meg, ez pedig a ***fedés (recall)***. Ezt a két jellemzőt egy ún. ***F-mérték*** foglalja össze, ami a pontosság és fedés súlyozott harmonikus közepe, ez $F_{\beta=1}$ esetén egyenlő súllyal veszi figyelembe a pontosságot és a fedést.

A kiértékelési módszer bevezetése lehetőséget biztosít a szerzett tapasztalatok visszacsatolására. Az optimalizálhatóság érdekében paraméterezhetővé tettük a komplex módszert és a szimulált hűtés algoritmusa alapján készített kereteljárással úgy optimalizáljuk a modellkészítés paramétereit, hogy a felismerési pontosság maximális legyen.

Szintaktikai elemzési módszerek alkalmazásai magyar nyelvre

A szerző felkészítette és kiértékelte a szintaktikai elemzők többféle változatát magyar nyelvű szövegeken a Szeged Treebank [Csendes05] adatait felhasználva, valamint alkalmazta különféle természetes nyelvvel kapcsolatos feladatokra készült összetett rendszerekben.

A magyar nyelv számos olyan nyelvi sajátossággal rendelkezik, ami megnehezíti a szintaxisfelismerést az indoeurópai nyelvekhez (pl. angolhoz) képest. Az egyik jelentős különbség a viszonylag szabad szórend, ami az igei vonzatkeret elemeinek átrendezhetőségét jelenti. A mondatrészi szerepet a magyar nyelv nem szórenddel fejezi

ki, hanem ragozással és névutók alkalmazásával oldja meg. Ebből adódik a másik probléma, a nagyfokú morfológiai változatosság. Az említett sajátosságok összességében jelentősen megnövelik a lehetséges minták, nyelvi sémák számát, melyek rontják a statisztikai alapú gépi tanulás hatékonyságát.

A szintaktikai elemzés leggyakrabban előforduló és egyik legfontosabb egysége a ***főnévi csoport (NP)***, mely általában névelővel kezdődik és főnévvel végződik, ez utóbbit az NP fejének is nevezünk. Ha nem lennének ez alól kivételek az NP-k felismerése nagyon pontos lehetne, azonban névelő bizonyos esetekben elhagyható, bizonyos esetekben viszont nem:

[$_{NP}$ *Péter*] [$_{NP}$ ~~*(egy)*~~ *könyvet*] *olvas* .

[$_{NP}$ *Péter*] *olvassa* [$_{NP}$ *a könyvet*] .

Ha a kontextus ezt lehetővé teszi, az NP feje is hagyható, tehát ez alapján előfordulhat olyan NP is melynek az utolsó szava nem főnév:

[$_{NP}$ Péter] [$_{NP}$ a régi könyvet] olvassa , [$_{NP}$ Mari] pedig **[**$_{NP}$ az **újat]** .

A mondatok szintaktikai szerkezetét leíró, ún. treebank reprezentáció a legtöbb nyugat-európai nyelvre, de számos közép-, ill. kelet-európai nyelvre már létezik, ezért időszerűnek bizonyult egy morfológiai és szintaktikai annotációt tartalmazó magyar nyelvű treebank létrehozása is. A Szeged Treebank kialakításakor a magyar nyelvre már ismert forrásmunkákra és meglévő elméletekre támaszkodva nyelvész szakértők egy konzisztens szintaktikai szabályrendszert dolgoztak ki. A treebank kidolgozása több munkafázisban történt és az adott állapot információtartalma meghatározta az ez alapján készült szintaktikai elemző felhasználási lehetőségeit. A treebank első verziója főnévi csoportok felismerését végző elemzők felkészítését tette lehetővé.

A ***felszíni szintaktikai elemzés (Shallow Parsing)*** során nem törekszünk arra, hogy feltárjuk a teljes szintaxist és ez olyan egyszerűsítésekre ad lehetőséget mely által az elemzési fázis felgyorsítható és a felismerés pontossága is javítható. Ilyen leegyszerűsített feladat a ***legbelső/legkülső főnévi csoportok*** (***base-NP/top-NP***) határainak meghatározása. A szerző által megvalósított felszíni elemző [Hócza04a] általános és üzleti szövegeken volt felkészítve és kiértékelve.

Felszíni elemzés esetén a komplex módszerben a tanulás és a felismerés leegyszerűsödik, valamint lehetőség nyílik a helyzet kihasználására speciális módszerek alkalmazásával. Mindez javítja a hatékonyságot, azaz gyorsabb és pontosabb eredményt kapunk, mintha a teljes szintaktikai elemzés eredményéből nyernénk a szócsoportokat. Például nem kell a teljes mondatot elemezni, a mondat egy címkéző algoritmussal kisebb részekre szegmentálható és a faminták illesztését csak a szócsoportok jósolt határain belül kell elvégezni.

Számos olyan alkalmazás van, ahol elegendő a szövegek felszíni szintaktikai elemzése. Ilyen például az ***automatikus információkinyerés (Information Extraction)*** vagy a ***szöveg kivonatolás (Text Summarisation)*** is. A szerző és társai által készített információ kinyerő rendszer [Hócza03b] a szövegek feldolgozásának különféle fázisait megvalósító moduljait láncszerűen összekapcsolva (***toolchain***) működik. A rendszer bemeneteként kapott egyszerű szövegfájlból az egymásra épülő részelemzések automatikus végrehajtásával előállítja a kinyert információkat tartalmazó strukturált adatbázist, eközben a rendszernek a következő részfeladatokat kell megoldania: mondat- és szószegmentálás, nyílt tokenosztályok és tulajdonnevek felismerése, morfológiai elemzés, szófaji egyértelműsítés, felszíni szintaktikai elemzés, szemantikus keretek illesztése és a felismert információk átírása strukturált adatbázisba. A rendszert üzleti híreket tartalmazó szövegekre alkalmaztunk.

A teljes szintaktikai elemzés több szempontból nehezebb probléma mint a felszíni elemzés, mivel sokféle szócsoport van és ezek mélyebb, összetettebb szerkezeteket alkotnak, emiatt a tanulás több mintát állít, valamint (a felszíni elemzéssel ellentétben) teljes szintaxisfát kell építeni. De a legnagyobb problémát magyar nyelv esetén az igei vonzatkeret modellezése jelenti, mivel a vonzatkeret elemek szabadok átrendezhetőek és nem feltétlenül összefüggősége mondatrészt alkotnak, emiatt ezt a jelenséget generatív jelegű szabályokkal nem lehet hatékonyan ábrázolni.

A szerző az általa kifejlesztett famintákon alapuló teljes szintaktikai elemzőjét a Szeged Treebank 2.0 adataiból vett általános szövegek és üzleti híreken készítette fel és értékelte ki [Hócza06a]. A szerző és társai a megtanult faminták felismerési pontosságának javítására alkalmazták a Boosting algoritmust [Hócza05a]. A [Hócza05b] cikkben a szerző és társai beszámolnak arról, hogy kialakítottak a Szeged Treebank 2.0 állományaiból egy mintaadatbázist és javasolták, hogy az eddig elkészült és az ezután kifejlesztett magyar nyelvű szintaktikai elemzők a pontos összehasonlíthatóság érdekében ezen legyenek felkészítve és kiértékelve.

A ***gépi fordítás (Machine Translation)*** feladata egy adott természetes nyelven elkészült szöveg automatikus átfordítása egy másik természetes nyelvre. Manapság a legjobb megoldást a ***statisztikai gépi fordító (Statistical Machine Translation)*** rendszerek adják. A szerző megvalósította egy ilyen rendszer, GenPar kiegészítését úgy, hogy beépítette magyar-angol nyelvpárt [Hócza06b], azaz egy inputként beadott magyar szövegnek a rendszer outputjaként megkapjuk az angol nyelvű fordítását. A rendszer tulajdonságainak feltérképezése céljából több prototípus is készült. A rendszerben szereplő magyar szövegek elemzéséért felelős modulok (szófaji egyértelműsítő és teljes szintaktikai elemző) a Szeged Treebank annotált szövegein voltak felkészítve, az angol nyelvért felelős rész pedig a rendszerrel adott mintaprototípusok részeként adottak

voltak. A GenPar betanításához és a kiértékeléshez szükség volt még párhuzamos mondatokra, azaz magyar nyelvű mondatokhoz rendelt angol fordításra. Ezeket a mondatpárokat a Hunglish Corpus [Varga05] adattárából választottuk ki, 5 ezer tanító és 500 teszt mondatpárt.

A disszertáció tézisei

A szerző értekezésben beszámolt az elmúlt években elért tudományos eredményeiről. Ezek két csoportba oszthatók, egyrészt beszélhetünk elméleti konstrukciókról és gyakorlati alkalmazásokról. Az első csoportba sorolhatóak a következő elméleti eredmények:

I/1. A szerző kidolgozott egy új formalizmust, melyet famintáknak nevezett el [Hócza04a]. A faminták mondatokon belül nagyobb, több szintű szintaktikai egységeket különítenek el, ugyanakkor hasonló szerkezetek összevonására is lehetőséget biztosítanak, így hatékony eszközt adnak az olyan ragozó és szabad szórendű nyelvek elemzéséhez, mint például a magyar nyelv.

I/2. A szerző kifejlesztett egy általános mintatanuló algoritmust, mely az RGLearn nevet kapta [Hócza04a]. Az algoritmus megkeresi a minták általánosítása és specializálása közötti optimális arányt, így famintákra alkalmazva azt a faminta halmazt, amely a maximális pontosságú szintaktikai elemzést adja.

I/3. A szerző elkészítette a chart parser szintaktikai elemző algoritmus famintákra alkalmazható változatát, mellyel bottom-up elemzés végezhető [Hócza04a].

I/4. A szerző egy komplex faminta alapú szintaktikai elemző módszerbe foglalta össze az egyedi lépéseket: kiinduló mintahalmaz gyűjtése, tanulás, elemzés, kiértékelés, modell optimalizálás [Hócza04a].

A gyakorlati alkalmazások az alábbi pontokba foglalhatók össze:

II/1. A szerző elkészített egy szövegkörnyezeti mintákon alapuló szófaji egyértelműsítőt, melynek alkalmazható mintáit az RGLearn algoritmussal állította elő. A módszer összehasonlításra került a szerző társai által kidolgozott módszerekkel [Kuba04].

II/2. A szerző alkalmazta a komplex faminta alapú módszert magyar nyelvű szövegek főnévi csoportjainak tanulására és felismerésére [Hócza04a]. Az

szerző által elért eredmények jelentős javulást mutattak a magyar nyelvre ezt megelőzően közölt eredményekhez viszonyítva.

II/3. A főnévi csoportokra alkalmazott felszíni elemzés beépítésre került a szerző és társai által készített információkinyerő rendszerbe [Hócza03b], mely magyar nyelvű gazdasági rövidhíreken volt felkészítve és kiértékelve.

II/4. A komplex faminta alapú módszert a szerző alkalmazta magyar nyelvű szövegek teljes szintaktikai elemzésére [Hócza05b], [Hócza06a].

II/5. A szerző és társai a teljes szintaktikai elemzés faminta tanuló modelljét a Boosting algoritmussal optimalizálták [Hócza05a].

II/6. A szerző a teljes szintaktikai elemzőt beépítette a GenPar gépi fordító rendszerbe és létrehozott egy új, magyar-angol fordításra alkalmas kiegészítést [Hócza06b].

Hivatkozások

[Aarts89] E. H. L. Aarts, E., Korst, J. (1989): Simulated Annealing and Boltzmann Machines, *John Wiley & Sons*, New York

[Baker79] Baker, James K. (1979): Trainable grammars for speech recognition, in *Proceedings of the Spring Conference of the Acoustical Society of America*, pp. 547–550.

[Black91] E. Black, S. Abney, D. Flickenger, C. Gdaniec, R. Grishman, P. Harrison, D. Hindle, R. Ingria, F. Jelinek, J. Klavans, M. Liberman, M. Marcus, S. Roukos, B. Santorini and T. Strzalkowski (1991): A procedure for quantitatively comparing the syntactic coverage of English grammars, in *Proceedings of the DARPA Speech and Natural Language Workshop*, pp. 306-311.

[Charniak93] Charniak, E (1993): Statistical Language Learning, *MIT Press*, Cambridge, Massachusetts

[Csendes05] Csendes, D., Csirik, J., Gyimóthy, T., Kocsor, A. (2005): The Szeged Treebank, in *Proceedings of the 8th International Conference on Text, Speech and Dialogue, TSD 2005*, Karlovy Vary, pp. 123-131

[Hócza03b] Hócza, A., Alexin, Z., Csendes, D., Csirik, J., Gyimóthy, T. (2003): Application of ILP methods in different natural language processing phases for

information extraction from Hungarian texts, in *Proceedings of the Kalmár Workshop on Logic and Computer Science*, Szeged, pp. 107-116.

[Hócza04a] Hócza, A. (2004): Noun Phrase Recognition with Tree Patterns, in *Acta Cybernetica*, Szeged, Volume 16, Issue 4, pp. 611-623

[Hócza05a] Hócza, A., Felföldi, L., Kocsor, A. (2005): Learning Syntactic Patterns Using Boosting and Other Classifier Combination Schemas, in *V. Matousek et al. (Eds.): Proceedings of the 8th International Conference on Text, Speech and Dialogue, TSD 2005*, TSD 2005, Karlovy Vary, Czech Republic, LNAI 3658, pp. 69-76

[Hócza05b] Hócza, A., Kovács, K., Kocsor, A. (2005): Szintaktikai elemzők eredményeinek összehasonlítása, *MSZNY 2005 konferenciakiadványa*, Szeged, 277-284 oldal

[Hócza06a] Hócza, A. (2006): Learning Tree Patterns for Syntactic Parsing, in *Acta Cybernetica*, Szeged, Volume 17, Issue 3, pp. 647 - 659

[Hócza06b] Hócza, A., Kocsor, A. (2006): Hungarian-English machine translation using GenPar, in *Proceedings of the 9th International Conference on Text, Speech and Dialogue, TSD 2006*, Brno, Czech Republic, September 11-15, pp. 87-94

[Kuba04] Kuba, A., Hócza, A., and Csirik, J. (2004): POS Tagging of Hungarian with Combined Statistical and Rule-Based Methods, in *Proceedings of the 7th International Conference on Text, Speech and Dialogue TSD 2004*, Brno, Czech Republic, September 8-11, pp. 113-120

[Kaplan73] R. M. Kaplan (1973). A general syntactic processor. In Rustin, R. (Ed.), *Natural Language Processing*, pp. 193-241. Algorithmics Press, New York.

[Kay86] Martin Kay. (1986). Algorithm schemata and data structures in syntactic processing. In Readings in natural language processing, pp. 35-70. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[Prescher03] Prescher, D. (2003): A Tutorial on the Expectation-Maximization Algorithm Including Maximum-Likelihood Estimation and EM Training of Probabilistic Context-Free Grammars, *Presented at the 15th European Summer School in Logic, Language, and Information (ESSLLI 2003)*.

[Quinlan93] Quinlan, J. R. (1993): C4.5: Programs for Machine Learning, *Morgan Kaufmann Publisher*.