A frázisstrukturált Szeged Treebank átalakítása függőségi fa formátumra

Alexin Zoltán

Szegedi Tudományegyetem, Szoftverfejlesztés Tanszék,
H-6720 Szeged, Árpád tér 2.
e-mail: alexin@inf.u-szeged.hu

Abstract. A CoNLL (Conference on Computational Natural Language Learning) nemzetközi konferencia szervezői évről évre különböző versenyeket írnak ki a résztvevők számára. Az elmúlt években került sor, például a tagmondatokra bontás (2001), tulajdonnév felismerés (2003), szemantikus szerep annotáció (2005) témakörében feladatok kiírására. A 2007. nyarán Prágában megrendezésre került konferencia versenyfeladványa a függőségi struktúrák gépi tanulása volt. A kitűzött feladatok között a magyar Szeged Treebank 2.0-ból kialakított tréning adatbázis is szerepelt. A versenyben egymástól nyelvészetileg rendkívül eltérő nyelvekre készített adatbázisok vettek részt (arab, baszk, katalán, kínai, cseh, angol, görög, magyar, olasz, török), amelyek 9 nyelvcsaládból származtak (sémi, elszigetelt, újlatin, kínai-tibeti, szláv, germán, hellén, finn-ugor, török). A szerző a függőségi fa formára történő automatikus gépi átalakítást mutatja be, valamint a verseny eredményeként kapott néhány megállapítást a nyelvcsaládokra vonatkozóan.

1. Bevezetés

A CoNLL 2007 konferencia szervezői megkeresték a világ különböző országaiban működő kutatókat, nyelvi korpuszok fejlesztőit, hogy nyújtsanak segítséget egy a függőségi struktúrák gépi tanulása témában kiírandó verseny feladatainak kialakításában. Nyújtsanak segítséget egy tréning és egy teszt adatbázis kialakításában. A kívánt tréning adatbázis mérete 50-100 ezer token, a teszt adatbázis mérete 5-10 ezer token volt. A magyar Szeged Treebank 2.0 [1] készítői is kaptak egy ilyen felkérést. A szervezők megkülönböztetett érdeklődést mutattak egy új, struktúrájában merőben más, eddig számukra ismeretlen nyelv iránt. A Szeged Treebank azonban a mondatok elemzését frázisstrukturált formában tartalmazza, amelyet egy szűk időkeretben függőségi fa formára kellett átalakítani.

A frázisstrukturált korpuszban a mondatok tagmondatokból felépülő hierarchikus struktúrát alkotnak. Maguk a tagmondatok pedig igékre, az igék vonzataira (ezek névszói szerkezetek) és egyéb alkotóelemekre bonthatók, amelyek az egyes szinteken belül azonban nem alkotnak hierarchiát. A függőségi fa formátum ettől abban tér el, hogy minden egyes szó a mondatban szigorúan egy másik szó alárendeltségében van. A mondatfa csúcsán egy mesterséges gyökér elem (ROOT) található, amelynek alá-

rendeltjei lesznek a mondatban előforduló szavak. A függőségi fában szereplő kapcsolatokat címkékkel is ellátják, amelyek a kapcsolat jellegére utalnak.

A verseny számára a Népszabadság és a HVG folyóiratok egy-egy számából kialakított korpuszrészletet választották ki, amely méretben megfelelt az elvárásoknak és a nyelvezete is kellően stabil volt. A verseny céljaira kialakított függőségi fa korpuszt a szervezők által biztosított segédprogramokkal ellenőrizték.

A szervezők a verseny eredményeinek összefoglalását egy hosszú tanulmányban tették közzé [2]. A résztvevők különböző gépi tanuló algoritmusokat használtak. A kapott eredmények azonban azt mutatták, hogy nem annyira az alkalmazott módszer, hanem az egyes nyelvcsaládok jellegzetességei, és az adott korpusz határozza meg a tanulás sikerességét. Így a nyelveket a tanulhatóság szempontjából három csoportba lehetett sorolni. A magyar nyelv a középső osztályba került, ami fontos visszajelzés arról, hogy a treebankben az igei vonzatok annotációja jó minőségű és releváns információt hordoz, illetve, hogy a konverzió algoritmusa kellően stabil. Ez lehetőséget adhat arra, hogy e munka eredményeként a teljes Szeged Treebank 2.0-át automatikus módszerekkel függőségi fa alakúra konvertáljuk.

2. A konverzió

A Szeged Treebank tartalmazza a mondatokban szereplő igék vonzatainak megjelölését, valamint egy a kapcsolat jellegére utaló címkét is. Az átalakítás fő feladata az volt, a vonzatokban kódolt függőségeket a konvertáló program vonja ki az adatbázisból, a nem jelölt függőségeket pedig automatikusan határozza meg.

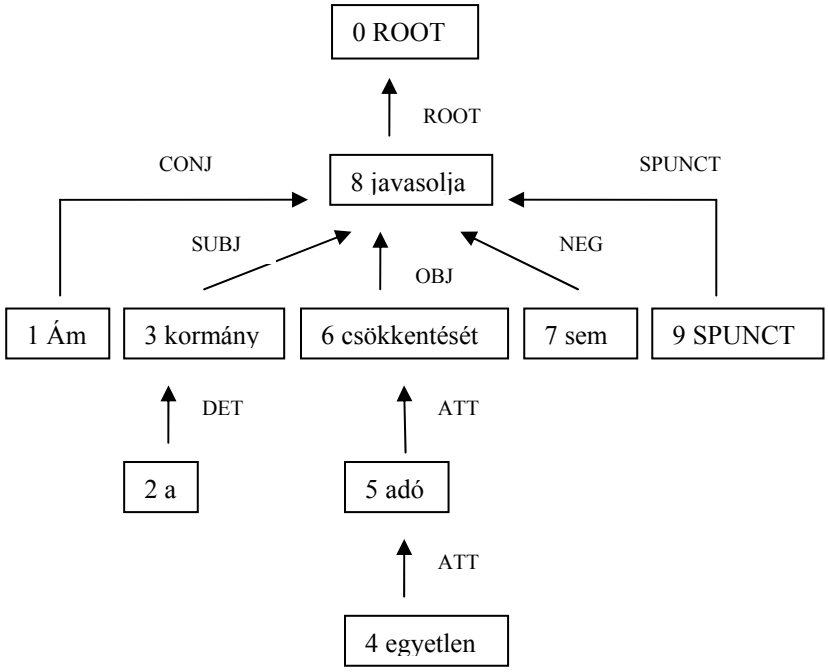
```
1       Ám      ám      C       Cc      ctype=coordinating      8       CONJ
2       a       a       T       Tf      def=yes 3       DET
3       kormány kormány N       Nc
                        n=singular|case=nominative|proper=no 8       SUBJ
4       egyetlen        egyetlen        A       Af
        deg=positive|n=singular|case=nominative     5       ATT
5       adó     adó     N       Nc
        n=singular|case=nominative|proper=no 6       ATT
6       csökkentését    csökkentés      N       Nc
        n=singular|case=accusative|proper=no|pperson=3rd|pnumber=singula
r       8       OBJ
7       sem     sem     R       Rm      _       8       NEG
8       javasolja       javasol V       Vm
        mood=indicative|t=present|p=3rd|n=singular|def=yes  0       ROOT
9       .       _       SPUNCT  SPUNCT  _       8       PUNCT
```

1. ábra. A tréning adatfájl egy részlete (HVG.2.6.4 mondat)

A treebankben az egyes névszói szerkezetek belső felépítése részben hierarchikus (az egymással birtokviszonyban álló névszók esetén), az egyes névszókhoz tartozó névelő, számnevek és jelzők viszont a hierarchia jelölése nélkül szerepelnek a frázis fejében lévő kulcs névszóval, alapvetően egy főnévvel közös szerkezetben. A főnevek bővítményeit a konvertáló program automatikusan a főnév alárendeltségébe tette. Névutós szerkezet esetén a konvertáló program a névszót tette a névutó alárendeltségébe – ez volt a legegyszerűbb és leginkább kézenfekvő megoldás.

Az algoritmus számára a legnehezebb feladat az volt, amikor egy tagmondaton belül több igei szerkezetet is talált, és a vonzatok felváltva tartoztak az egyes igékhez (igenevekhez) valamint az igék (igenevek) között is függőségi kapcsolat állt fent. Az elkészített program három igei szerkezetet tud kezelni egy tagmondatban, ami a tesztek alapján elegendő volt.

Az 1. ábrán látható a konvertált állomány egy kis részlete. A mondat szavai 1-től kezdve egy-egy sorszámot kapnak. A ROOT elem a 0-s sorszámot kapja. A táblázatos (tabulátor karakterekkel tagolt fájl) egyes oszlopai a következők: sorszám, ortográfia, szótári alak, a morfo-szintaktikai kód (MSD) első betűje, részletes morfo-szintaktikai kód (a verseny szervezői kérték, hogy az MSD kódokból itt csak az első két karakter szerepeljen), további fontos lexikális tulajdonságok | (bar) karakterrel elválasztva, a fölérendelt szó sorszáma, a függőségi kapcsolat jellege.



2. ábra. A konvertáló program által előállított függőségi fa

A konverziót megvalósító program C# programozási nyelven készült és alapvetően heurisztikus eszközökkel oldotta meg a feladatot. Az idő rövidsége miatt került sor ennek az alkalmazására az XSLT transzformációs eszköz helyett. A program tagmondatonként rekurzívan végezte el a konvertálást. Első lépésben megszámolta a tagmondatban szereplő igék (igenevek) számát. Ha volt ige, akkor megkereste a hierarchiában legfelsőt, és elhelyezte a többi igét (igenevet) ez alá, majd pedig végighaladt a bővítményeken és ezeket is a megfelelő ige alá helyezte el. A gazdátlan bővítmények a legfelső szintű ige alárendeltjei lettek. Főnévi szerkezetek esetén a jobb oldali

első főnevet (fejet) tette az algoritmus a legfelső pozícióba és ez alá helyezte el a frázis további elemeit. A névelő függőségi címkéje DET, a jelzőké ATT lett.

A rendszerben természetesen előfordulhattak hibák. A rövid 2-3 hetes határidő miatt nem volt mód minden hibajelenség okát felkutatni, ezért a rendszer a szerkezeti hibás mondatokat egy belső ellenőrző eljárással észlelte és az output állományból egyszerűen törölte, ezek száma azonban nem volt túl sok: HVG: 9 mondat (2179-ből), Népszabadság 38 mondat (3905-ből).

3. Eredmények

A konvertált tréning és teszt állományt a szervezők a verseny résztvevőinek kiadták és 21 csapattól érkezett eredmény a magyar nyelvre. Nagyon sok különböző módszert alkalmaztak: pld. SVM, véges Newton SVM, maximum entrópia, átlagolt perceptron, maximum likelihood, HMM alapú módszereket stb. a függőségi relációk predikciójára. A kapott eredmények azt mutatták, hogy nem annyira az alkalmazott módszer, hanem az egyes nyelvcsaládok jellegzetességei és az adott korpuszok határozzák meg a tanulás sikerességét. Így a nyelveket a tanulhatóság szempontjából három csoportba lehetett sorolni az elért átlagos pontosság alapján. Nehezen tanulható (76,31-76,94%): arab, baszk, görög. Közepesen jól tanulható (79,19-80,21%): cseh, magyar, török. Jól tanulható (84,40-89,61%): katalán, kínai, angol, olasz.

Az eredmények elemzése azt mutatta, hogy ez az eredmény sok esetben a korpusz nagyságával és minőségével van kapcsolatban, hiszen az arab és az újgörög nem annyira nehéz nyelvek. Mégis megelőzte őket a magyar és a cseh, amelyek nagymértékben ragozók és meglehetősen szabad szórendet engednek meg. A szervezők megvizsgálták az ismeretlen szavak arányát a teszt állományokban, amely érték a magyar és a török nyelvek esetén volt a legmagasabb. Ez nem rontotta le a magyar eredményt – a sok ismeretlen, új szót ellensúlyozni tudta a korpusz mérete.

A verseny eredményeit az [2] irodalom foglalta össze, amelyből kitűnik, hogy a heurisztika ellenére a magyar függőségi treebankkel kapott eredmények a középmezőnyben foglaltak helyet. Az elkészített program nemcsak konvertálására alkalmas, hanem arra is, hogy a Szeged Treebankben meglevő annotálási hibák után nyomozzon, segítsen ezek felkutatásában és kijavításában. A jövőben szeretnénk ezt a lehetőséget is felhasználni a treebank minőségének javítására.

4. Irodalom

[1] Csendes D., Csirik J., Gyimóthy T., Kocsor A.: The Szeged Treebank, in Proceedings of the Eighth International Conference on Text, Speech and Dialogue (TSD 2005), Karlovy Vary, Czech Republic 12-16 September, and LNAI series Vol. 3658, pp. 123-131 (2005)

[2] Joakim Nivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, Deniz Yuret: The CoNLL 2007 Shared Task on Dependency Parsing, in Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007, pp. 915–932, Prague, (2007)