Magyar társadalomtudományi citációs adatbázis: A MATRICA projekt eredményei

Váradi Tamás, Mittelholcz Iván, Blága Szabolcs, Harmati Sebestyén

MTA, Nyelvtudományi Intézet,
Benczúr utca 33., 1068 Budapest
e-mail: {varadi, mittelholcz}@nytud.mta.hu
{szabolcs.blaga, harsej}@gmail.com

Kivonat A szabad szövegekből történő strukturált információkinyerés egy sajátos területe a tudományos közlemények feldolgozása. Ezen belül is különösen fontos feladat a közleményekben szöveges alakban található hivatkozások kinyerése, elemzése és adatbázisba építése.[1] Ez röviden a célja a MATRICA (*Magyar Társadalomtudományi Citációs Adatbázis*) projektnek, ami a 2010-ben forráshiány miatt megszakadt HUN-ERIH projekt folytatása.[2] A projekt megvalósítása során, különösen a majdani felhasználókkal, az egyetemi könyvtárakkal való együttműködés eredményeként az alábbi prioritások alakultak ki: 1) tudományos cikkek feldolgozása a nyers fájloktól az adatbázisig, 2) ahol lehet, ott az emberi közreműködés csökkentése, 3) ahol kell, ott a szükséges emberi beavatkozáshoz kényelmes webes felület biztosítása.

1. Bevezetés

Napjaink tudományos életében a kutatókra és könyvtárosokra egyre nagyobb terhet ró a bibligráfiai adatok rögzítése és követése. Ezért is tartotta fontosnak az MTA egy olyan technológiai lánc kifejlesztését, ami alkalmas nagy mennyiségű, elektronikus formában elérhető cikk bibliográfiai adatainak feldolgozására; számítógépes eszközökkel támogatva meg az eddig jellemzően kézi munkával végzett hivatkozásfeldolgozást. A technológiai lánc mellett fontos, hogy a társadalom- és bölcsészettudományi területen Magyarországon még nem létezett egy átfogó bibliográfiai adatbázis, amely természettudományi területen már adott. Ennek létrehozása volt a Matrica projekt másodlagos célja.

2. Kapcsolódó munkák

Az interneten szabadon elérhető és kipróbálható hivatkozásfeldolgozó szoftverek[3] elsősorban az egyéni munka segítésére hivatottak: az egyes kutatók dolgát

[1] L. [2,3]
[2] A projekt előző szakaszáról l. [1].
[3] L. többek közt az alábbiakat:
cb2bib (http://www.molspaces.com/d_cb2bib-overview.php),

könnyítik meg a saját bibliográfiájuk összeállításában. A mi célunk két dologban tér el ettől. 1) Mivel nagy mennyiségű és heterogén publikáció feldolgozását tűztük ki, ezért nem fogadhattuk el a feldolgozás olyan fokú pontatlanságát, ami a személyes használatra szánt programokat jellemzi, mivel a kézi javítás ekkora mennyiségben már nem gazdaságos. Az általunk kezelt anyag hivatkozási konvencióinak heterogenitása miatt szintén nem volt céljainknak megfelelő egy olyan szabályalapú megközelítés, amely csak néhány hivatkozási sztenderdet képes kezelni. 2) Mivel alapvetően egy közös bibliográfiai adatbázis létrehozásában gondolkodtunk, elengedhetetlen volt a kollaboratív munka támogatása egy webes felületen keresztül.

3. Nyers hivatkozások kinyerése

3.1. Fájlok

A MATRICA projektben a HUN-ERIH alatt összegyűjtött anyagot örököltük, azt már külön nem bővítettük és nem frissítettük, csak a feldolgozására koncentráltunk. A HUN-ERIH projekt vállalása a kortárs, magyarországi (de nem feltétlenül magyar nyelvű) bölcsészet- és társadalomtudományi folyóiratok feldolgozása volt öt évre visszamenőleg. Igyekeztünk minél szélesebb körből meríteni, és a kiadókkal való egyeztetések után végül 192 folyóirattól sikerült anyagot szereznünk.

A folyóiratok rendelkezésünkre bocsátott állománya nagyon heterogén, mind a fájlok terjedelmét, mind azok formátumát tekintve.

A folyóiratok egy része minden cikket külön fájlban tárolt, másik része folyóiratszámonként, harmadik része évfolyamonként bontotta fájlokra az anyagot. Ez jelentősen megnehezítette a cikkek beazonosítását és a cikkekre vonatkozó metaadatok kinyerését: a fájlokat először feldaraboltuk cikkekre, azután az egyes cikkekből egyrészt az azok azonosításához szükséges úgynevezett fejléc adatokat, másrészt a cikkhez tartozó összes hivatkozás nyers alakját nyertük ki.

Ami a fájlok formátumát illeti, a hét különböző formátum közül a HTML (43%) és a PDF (51%) bizonyult a leggyakoribbnak. A HTML-fájlokhoz képest a PDF-állományok feldolgozása jelentős többletmunkával járt.

3.2. PDF feldolgozás

A PDF fájlok szerkezete nagyon egyszerű, alapvetően minden egyes karakter abszolút geometriai pozícióját adja meg egy adott hordozón (előre megadott méretű téglalap alakú területen – papíron). Az abszolút pozíció megadása egy kétdimenziós koordináta-rendszer segítségével történik, hasonlóan az egyes karakterek kiterjedéséhez. Ezen felül szerepel a karakter mérete, amely így nem feltétlenül tölti ki a számára megadott téglalap alakú területet, valamint az

text2bib (http://text2bib.economics.utoronto.ca),
Free Cite (http://freecite.library.brown.edu/welcome),
Simple Text Query (http://www.crossref.org/SimpleTextQuery/).

aktuális karakterkódolási táblázat szerinti kódja (ami megadja, hogy milyen betű jelenjen meg vizuálisan), illetve a használt betűtípus. Karakternél magasabb rendű szövegbeli egységek (szó, sor, bekezdés, stb.) a karakterek csoportosításával jönnek létre. Ugyanakkor a legtöbb esetben a szöveg magasabb rendű felépítése nem megbízható (nem tükrözi a forrás struktúrális elrendezése a hordozón látható vizuális elrendezést), ezért a legtöbb esetben a szöveg struktúráját a karakterszintű elemek pozícióinak elemzéséből algoritmikusan kell rekonstruálni. További nehézséget jelent, hogy a PDF belső szerkezete jelentős rugalmasságot biztosít az előállításkor, így a különböző folyóiratok között szinte minden esetben, de akár a folyóiratok egyes számain belül is változhat a PDF belső szerkezete, attól függően, hogy milyen alkalmazással készítették az adott állományt. Bár a PDF belső szerkezete egységes keretet ad a dokumentumok felépítéséhez, mégis a különböző PDF-készítő programok más-más egyedi mechanizmus mentén nagyon eltérő belső struktúrájú fájlokat hoznak létre.

A belső szerkezet változásairól sok esetben a készítőnek sincs tudomása, ezért erről semmiféle analitikus információ nem áll rendelkezésünkre, tehát olyan általános feldolgozó eljárást kellett kialakítanunk, ami a PDF-fájlok egy nagyon diverz halmazára alkalmazható.

Az egyik jellemző probléma a többhasábos elrendezésű szövegek kezelése, itt sok esetben a különböző hasábokhoz tartozó azonos magasságban lévő szövegrészek egy sorként voltak tárolva PDF belső szerkezete alapján, így ezeknél a karakterek közötti térköz vizsgálatával kellett visszaállítani az eredeti többhasábos struktúrát.

Mivel egy adott PDF-állományban több cikk is szerepelhetett egyszerre, ezért a cikkek elhatárolásához és egy adott cikk metaadatainak megtalálásához olyan feltételrendszereket kellett kidolgozni, melyek egyértelműen beazonosítanak egy adott szövegrészt. A beazonosításhoz szükség volt a magasabb szövegbeli egységek helyes felismerésére, illetve a különböző formázási elemek egységes kezelésére. Itt kihívást jelentett a címben, szerzők nevének megadásánál és a hivatkozásoknál is előszeretettel használt ún. kiskapitális írásmód kezelése. Sokszor a kiskapitális írásmód PDF-en belüli megvalósítása azt jelentette, hogy a csupa nagybetűvel írt szövegben változott az egyes karakterek mérete, ez normál szöveggel, vagy esetenként egyszerű nagybetűs írásmóddal keverve nehezen kezelhető, körültekintő mérlegelést igényel a feldolgozó algoritmust paraméterezők részéről. Természetesen néhány esetben a többértelműség nem oldható fel algoritmikusan, vagy csak túlzott fejlesztési erőforrásigény mellett, ezért a manuális javítás a jobb megoldás.

További nehézséget jelent a PDF-állományok eltérő karakterkódolása. Mivel a PDF lehetővé teszi az egyes szövegrészek közötti eltérő kódolási táblák használatát, ezért ezek kezelése sokszor külön óvatosságot igényel. A legnehezebben azok az esetek kezelhetőek, mikor a karakterek kódolásából nem, csak az adott betűtípus neve és megjelenése alapján derül ki, hogy milyen karakterek vannak kódolva az adott szövegrészben. Mivel az állományokban lehetségesen használható betűtípusok száma nagyon nagy, ezért ezek az esetek is csak egyedileg, speciális cseretáblák segítségével, vagy manuális javítással kezelhetőek.

Mivel a PDF-ek belső szerkezete jelentős eltéréseket mutatott, ezért tűnt jó megközelítésnek egy lépésben megpróbálni olyan feldolgozót fejleszteni, ami minden lehetséges típusra megoldást kínál. A hatékony fejlesztés érdekében egyfajta evolúciós megközelítést használtunk, ami abból állt, hogy mindig visszavisszatérő módon fejlesztettük az algoritmusokat, hogy egyre nagyobb számú jelenséget legyenek képesek kezelni. A PDF-elemzés evolúciós fejlődése a feldolgozás előrehaladtával:

1. Dokumentumok elemzése, tipikus esetek kiválasztása.
2. A felmerült problémák kezelésére alkalmas elemző fejlesztése.
3. Az elkészült elemző alkalmazása minél többféle dokumentumtípusra.
4. Kimeneti pontatlanságok elemzése, elemző hibáinak feltárása.
5. Vissza az 1-es ponthoz.

A fejlesztési ciklusok során az egyik legfontosabb feladat annak eldöntése, hogy az adott probléma érdemes-e arra, hogy specifikus fejlesztést eszközöljünk az elemző programban, vagy hatékonyabb egyedi esetként kezelni, így spórolva a jelentős erőforrásigényű algoritmus fejlesztéssel a viszonylag ritkán előforduló „speciális" esetekben.

A feldolgozó fejlődésével párhuzamosan bővült a projektbe bevont csoportok köre, míg kezdetben csak a fejlesztői csapat dolgozott a problémákon, később a tesztelők és paraméterezők folyamatos bevonásával jelentős párhuzamosítást értünk el az egyes munkafázisokban és a csoportok egymás közti kommunikációja alapján minden csoport hatékonysága dinamikusan fejlődött. A kézi ellenőrzés jelenlegi szakaszban a nyers hivatkozások PDF-ekből való kinyerése 49,2%-os pontosságot mutat.

Az evolúciós fejlesztési ciklusok során fontos szempont a visszafelé kompatibilitás megőrzése, vagy az annak elvesztéséből származó munkaterhelés minimalizálása, ebben a tekintetben is egyensúlyra törekedtünk. Míg kezdetben gyorsan változott a feldolgozó program, a munka kiterjesztésével párhuzamosan a stabilitás is egyre fontosabbá vált.

A citációs adatbázis jövőbeni fejlődése és fenntarthatósága szempontjából jelentős előrelépés lenne, ha az egyes kiadók és szerkesztők olyan metainformációkkal látnák el kiadványaik elektronikus változatát, ami megkönnyíti az automatikus feldolgozást. Még jobb lenne, hogyha ez a formátum egységes lenne az egyes kiadványok között. A Matrica adatbázisba bekerülő cikkek esetében már bármilyen kimeneti formátum előállítható a későbbiekben.

4. Hivatkozások elemzése

A különféle formátumú fájlok feldolgozása és a nyers hivatkozások kinyerése után a következő lépésben ezen hivatkozások feldolgozása történik. A HUN-ERIH projekt során erre a célra a NooJ szoftvercsomagot[4] használtuk, amely lokális grammatikákat használ az egyes hivatkozáselemek (szerző, cím, kiadó stb.) felismerésére, majd ezek megfelelő kombinációit illeszti a hivatkozások különféle típusaira.

[4] http://www.nooj4nlp.net/pages/nooj.html

Ezzel a szabályalapú módszerrel meglehetősen alacsony F-mértékeket kaptunk egy kismintás kiértékelés során, valamint nem bizonyult elég robusztusnak a rendkívül heterogén adathalmazon. (A rendszer leírását és az eredményeket lásd az [1] cikkben.) Ezért döntöttünk úgy, hogy a projekt folytatásában statisztikai alapú gépi tanuló megoldást alkalmazunk. A maximum entrópián alapuló HunTag[5] rendszert választottuk, amelyet eddig főnévi csoportok ([4]) és tulajdonnevek ([5]) felismerésére használtak, de bármilyen szekvenciális címkézési feladatra alkalmas, így a hivatkozások parszolására is.

4.1. Az adathalmaz

A hivatkozások hasznos bibliográfiai adatmezőinek definiálásához a BibTeX szabványt vettük alapul, és az alábbi tizenkilenc mezőt határoztuk meg: szerzők, szerkesztők, cím, kötetcím, sorozat, kiadás, kiadás helye, folyóirat, kiadó, iskola (téziseknél), szervezet (konferenciáknál), intézmény (egyéb esetben), év, hónap, kötet, szám, oldalszám, megjegyzés (pl. ki fordította) és URL. Ezeken felül további öt olyan mezőt használunk, amelyeket a hivatkozások lényegi információt nem hordozó, de valamilyen pozíciót jelző elemeinek tartunk fent, mint például a szerkesztőket jelző *szerk.*, *eds.* vagy éppen *hrsg.* Hasonló mezőket definiáltunk a folyóiratszámokat és évfolyamokat jelző bibliográfiai elemeknek (pl. *vol*, *num*) és az oldalszámoknak (pl. *o.*, *p.*) is.

Tanítás és tesztelés céljára egy 12.000 hivatkozást tartalmazó mintát választottunk ki véletlenszerűen. A minta kézzel való felcímkézését diákok végezték, amit szakértő könyvtárosok ellenőriztek. Ezt az adathalmazt utólag kézzel szűrtük, hogy még tisztább tanító és kiértékelő anyaghoz jussunk, így egy kb. 10.000 hivatkozást tartalmazó gold standard korpuszhoz jutottunk. Ezt használtuk 80%/20%-os vágásban tanításra és kiértékelésre.

4.2. Jegykinyerés

A tanítás során a legfontosabb sztring értékű felszíni jegyek (karakter n-gram, a token n karakterből álló előtagja és utótagja) optimális kombinációját a teljes paramétertér bejárásával állapítottuk meg. Minden paraméterkombinációt ötszörös keresztvalidációval kimértünk, és az összesített F-mértékek alapján az 1-es n-gram, 5-ös előtag, 3-as utótag jegykombináció bizonyult a legjobbnak. Az 1-es n-gram rendre jobb teljesítményt nyújtott a többi felszíni jegy eltérő értékei mellett is, ezért elfogadtuk. A tanításhoz felhasználtunk városok, kiadók és folyóiratok neveit tartalmazó listákat is.

4.3. Kiértékelés

A kiértékelést a fent leírt gold standard adathalmazon végeztük, ötszörös keresztvalidációt alkalmazva. A táblázatban látható eredmények azt mutatják, hogy a

[5] https://github.com/recski/HunTag/

gyakori (és egyben fontos) mezők F-mértéke általában 90% felett van, míg a ritkán előforduló mezők várható módon rosszabb eredményt adnak.

mező	**pontosság**	**fedés**	**F-mérték**
szerzők	96,93	97,57	97,24
szerkesztők	91,60	91,56	91,58
cím	88,50	88,06	88,25
kötetcím	71,04	73,33	72,17
sorozat	31,91	28,86	30,31
kiadás	61,54	57,66	59,53
kiadás helye	92,02	91,37	91,69
kiadó	83,09	85,72	84,39
intézmény	53,01	54,63	53,81
szervezet	12,00	9,38	10,53
iskola	42,39	34,51	38,05
folyóirat	86,74	90,49	88,57
kötet	68,23	78,34	72,94
szám	75,62	70,12	72,77
év	97,67	94,30	95,95
hónap	65,26	55,11	59,76
oldalszám	95,79	95,10	95,44
megjegyzés	70,81	61,80	66,11
url	83,57	80,09	81,71
összesített	88,81	88,33	88,57

Külön említést érdemel két mezőcsoport: egyrészt az évfolyam és szám, másrészt a intézmény–szervezet–iskola hármas. Mindkét csoport esetében hasonló környezetekben alternáló címkékről van szó. Folyóiratok esetében nem ritka, hogy az évfolyam és a szám közül csak az egyiket adják meg, pl.

> Baumrind, D. (1978): *Parental disciplinary patterns and social competence in children.* Youth and Society. 9. 239–276.

Ebben az esetben a *9* az évfolyam és a szám is lehet, a rendelkezésre álló kontextus alapján nem derül ki egyértelműen, hogy melyik.

Hasonló a helyzet a kiadói pozícióban álló mezők esetében is; ezek: a tézisek kiadói (iskolák), a konferenciakötetek kiadói (szervezetek) és az egyéb, publikációt megjelentető, de kiadónak nem tekintett intézmények. Ezek a mezők túl azon, hogy azonos pozícióban szerepelnek, hasonló (intézmény)neveket is tartalmaznak, ami jelentősen megnehezíti a megkülönböztetésüket, nem csak a gépi tanuló algoritmus, hanem az annotátorok számára is. Ebből kifolyólag már a gold standard adathalmazban sem egységes ezeknek a mezőknek a jelölése. Ezt a megkülönböztetést az indokolta, hogy a BibTeX sztenderd mezőihez igazodtunk, de a jövőben érdemes lenne ezeket összevonni egy intézmény jellegű mező alá.

A folyamat végén azokat a hivatkozásokat, amelyek egy előre meghatározott küszöbértéknél alacsonyabb valószínűségű mezőt tartalmaznak, utólagos ellenőrzésre ajánlja fel a rendszer. Ezzel két, külön forrásból származó hibatípust is ki tudunk küszöbölni. Egyrészt lehet maga a hivatkozás valamilyen szempontból különleges, ami miatt az elemző kimenete nem elég megbízható. Másrészt ha még az első lépésben nem megfelelően történt a nyers hivatkozás kinyerése (pl. folyó szöveg vagy csonka hivatkozás lett kibányászva), azt is jelezni fogja a rendszer a hivatkozás elemzésének alacsony valószínűségével.

5. Felület

A feldolgozás hatékony párhuzamosítása érdekében egy sokfelhasználós webes felület került kialakításra. A felület célja, hogy a gépi feldolgozás irányítása, ellenőrzése, a szükséges kollaborációs feladatok kivitelezése egy egységes keretben, felhasználóbarát módon mehessen végbe. A felület funkcionalitását négy felhasználói csoport szerint lehet felbontani:

1. A létrejövő citációs adatbázis jól struktúrált megtekintése és különböző keresési funkciók megvalósítása.
2. A szükséges kézi javítások és ellenőrzések elvégzése, az adatbázis minőségének javítása, szakértői csoportok bevonása a feldolgozás minőségének javítása érdekében.
3. Új adatok bevitele, az automatikus feldolgozás körén kívül eső folyóiratok hozzáadása az adatbázishoz.
4. Az automatikus feldolgozás paraméterezése a háttérben futó feldolgozási folyamatok és azok eredményének nyomon követése, elemzése.

A webfelület minden tekintetben igyekszik a mai kor elvárásai szerint megkönnyíteni a különböző felhasználói csoportok közös munkáját. Mivel a elemzési folyamatok jelentős erőforrásigénnyel bírnak, ezért az erőforrások optimális kihasználása érdekében egy aszinkron feldolgozási mechanizmus került megvalósításra, ahol az egy időben aktív felhasználók egy globálisan meghatározott erőforráskvótán osztoznak, így nagy terhelés mellett is elkerülhető a rendszer túlzott lelassulása, a felület válaszideje kielégítő marad.

6. Összefoglalás

Az elvégzett munka eredményeként olyan technológiai lánc állt elő, amely lehetővé teszi nagy mennyiségű, heterogén elektronikus szöveg bibliográfiai adatainak félautomatikus feldolgozását. Önálló fejlesztésünk a PDF-ek kezelését megkönnyítő szoftver, a statisztikai gépitanuló modul testreszabása és felkészítése a hivatkozások parszolására, valamint a kollaboratív webes felület. Munkánk másodlagos eredménye maga a folyamatos feltöltés alatt álló citációs adatbázis, amivel reményeink szerint könnyebbé tehetjük a kutatók és könyvtárosok ezirányú munkáját, hogy érdemi feladataikra jobban koncentrálhassanak.

Hivatkozások

1. Váradi T., Pintér T., Mittelholcz I., Peredy M.: Bibliográfiai hivatkozások automatikus kinyerése. In: Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2010), Szeged, Magyarország, 56-65, (2010).
2. Bergmark, D.: Automatic extraction of reference linking information from online documents. TR2000-1821 (2000)
3. Day, M.-Y., Tsai, T.-H., Sung, C.-L., Lee C.-W., Wu, S.-H., Ong, C.S., Hsu, W.-L.: A knowledge-based Approach to Citation Extraction. In: Proceedings of the IEEE International Conference on Information Reuse and Integration. (IEEE IRI 2005). Las Vegas, Nevada, USA. (2005) 50-55.
4. Recski G., Varga D.: A Hungarian NP-chunker. The Odd Yearbook, (2009)
5. Simon E. Approaches to Hungarian Named Entity Recognition. PhD disszertáció. BME, Budapest, (2013)