Magyar hadifoglyok adatainak orosz-magyar átírása és helyreállítása, és a szabadszöveges adatbázisok tulajdonságai

Sass Bálint, Mittelholcz Iván, Halász Dávid, Lipp Veronika, Kalivoda Ágnes

Nyelvtudományi Intézet, ELKH, MTA
{sass.balint,mittelholcz.ivan,lipp.veronika,kalivoda.agnes}@nytud.hu
david.peter.halasz@gmail.com

Kivonat Ebben a tanulmányban a magyar hadifoglyok adatbázisában lévő tulajdonnevek orosz-magyar átírásának módszerét és tanulságait mutatjuk be. Az adatbázisban a 682000 hadifogoly adatai cirill betűkkel leírva állnak rendelkezésre. Az adatok két körben szenvedtek torzulást: először, amikor az adatokat felvevő szovjet katona hallás utána leírta, majd mikor 60 év múltán szintén orosz anyanyelvűek manuális munkával digitalizálták az anyagot a kézzel írott kartonokról. Esetünkben nem szimpla átírásról van szó, hanem valójában az eredeti magyar szó helyreállításáról. Külön feladatot jelentett a helyeket leíró adatok adatmezőkre bontása. Szabályalapú algoritmusunkban szigorú és laza átírást, valamint közelítő keresést alkalmazunk, az átírást listákkal vetjük össze. Ha egyik módszer sem vezet eredményre, akkor a buta betűről-betűre átírást adjuk vissza. Eredmény: az adatok 77%-ához tudtunk helyes helyreállított alakot rendelni. Megfogalmazunk tanulságot a kézzel készült, korlátozatlan, szabadszöveges adatbázisok szükségszerű következetlenségéről; valamint arról, hogy egyedi adatnál, tanulóadat híján van létjogosultsága a szabályalapú módszereknek.
Kulcsszavak: hadifogoly, átírás, transzkripció, szabályalapú, szabadszöveges adatbázis

1. Bevezetés

Harminc évnek kellett eltelnie a rendszerváltás óta, hogy Magyarország megkaphassa a II. világháború végén elhurcolt 682000 magyar hadifogoly nyilvántartási adatait. 2019-ben született meg a megállapodás a Magyar Nemzeti Levéltár és az Orosz Állami Hadilevéltár között az adatok átadásáról, és még ebben az évben meg is érkezett az anyag a Magyar Nemzeti Levéltárba. Az adatokat két formában kaptuk meg: (1) az eredeti kézzel, cirill betűkkel írt nyilvántartó kartonok digitalizált (szkennelt) változata; (2) cirill betűs leirat, adatbázis, amely a nyilvántartó kartonokon szereplő, az egyes személyekhez köthető információkat tartalmazza.

A Levéltár célja, hogy az információkat online kereshető formában közzétegye, lehetővé téve, hogy a leszármazottak hozzájussanak a rokonaikról tudható

információkhoz, illetve, hogy általánosságban kutathatóvá tegye az anyagot a szakemberek és a nagyközönség számára.

Az adatbázis jellemzője, hogy keletkezési helyének megfelelően minden adat *cirill betűkkel* szerepel benne. A kereshetőség biztosítása érdekében fontos feladat tehát az anyag átültetése magyarra. Ezt a feladatot végeztük a Nyelvtudományi Intézetben, erről számol be ez a tanulmány. A munkálatokban a Levéltár, a Helion Kft. és a Nyelvtudományi Intézet működött közre.

2. A feladat

A feladat tehát az, hogy magyarra alakítsuk az eredeti cirill betűkkel[1] leírt adatokat.

Ковач Йожеф → Kovács József

1. ábra: Alappélda. A *Kovács József* cirill formája és a helyreállítandó magyar változat. Itt elegendő, ha minden cirill betűt egyszerűen a magyar megfelelőjére írunk át.

Az 1. ábrán vázolt egyszerűnek tűnő feladat nehézségét több tényező adja. Egyrészt az a nyelvi tény, hogy az orosz betűk illetve hangok nem egy az egyben felelnek meg a magyar betűknek és hangoknak. Másrészt az anyag az elmúlt évtizedek során több alkalommal is torzult. A szovjet hadifogolytáborba érkező katonák általában nem rendelkeztek iratokkal (Katona és Szikla, 2014). Ennek megfelelően az adataikat legtöbbször *hallás után* írta le az adatokat felvevő szovjet katona. Az első torzulást tehát az okozza, hogy hallás után került rögzítésre a leíró által nem értett magyar nyelvű adat. A második torzulás akkor keletkezett, mikor az elmúlt években az Orosz Állami Hadilevéltár munkatársai manuális munkával digitalizálták a sokszor nagyon nehezen olvasható szkennelt kézírásos kartonokat, azzal együtt, hogy ők sem *értették* a leírtakat, sokszor csak a cirill betűsorozatokat igyekeztek beazonosítani és rögzíteni. A fenti két torzulás biztosra vehető, ezenkívül még lehetséges egy harmadik is: a kartonok valószínűleg nem közvetlenül a táborokban készültek, hanem egy központi helyen, így még egy másolási lépés beiktatódhat.

A történeti szövegeknél látunk hasonló jelenséget, mikor a hibázások és többszöri másolás eredményeképpen jelentősen meg tud változni az eredeti szöveg (Haader, 2014). Munkánk előzményének tekinthető Prószéky és mtsai (2002) cikke, mely a különböző forrásokból eredő karakterhibák javításával foglalkozik.

[1] A cirill betűk, az orosz ábécé és hangkészlet ismeretét feltételezzük a továbbiakban. Ld. pl. `https://hu.wikipedia.org/wiki/Orosz_ábécé`

3. Fordítás, átírás, helyreállítás

Ebben a részben az alapfogalmakat világítjuk meg: az elvégzendő feladat nem fordítás, nem egyszerű átírás (Bradley, 2020), hanem valójában – nevezhetjük így – helyreállítás.

	orosz	→ **magyar**
fordítás	конец	vége
átírás	конец	kányec
helyreállítás	Ковач	Kovács

1. táblázat. A helyreállítás viszonya a fordításhoz és az átíráshoz.

Ahogy az 1. táblázatban látni fogjuk, fordítás során a forrásnyelvű, forrásnyelvi írással írt szöveget tesszük át célnyelvű, célnyelvi írással írt szöveggé; átírás során ugyanezt, a forrásnyelvet megtartva célnyelvi írással átírt szöveggé; a helyreállítás során viszont az eleve célnyelvű, viszont forrásnyelvi írással írt szöveget alakítjuk a célnyelvet megtartva írásában is célnyelvivé. A lényeg tehát, hogy a szóban forgó feladatban – bár cirill betűkkel van leírva – a kiinduló elem egy *magyar* szó, ezt kell az eredeti magyar formájában és értelmében helyreállítani. Nemcsak át kell írni, hanem rá is kell jönni, hogy mi az. A feladat így jóval nehezebb, mint az egyszerű átírás, ahol a szó értelmére nem kell figyelmet fordítani.

4. A helyreállítás szintjei

Az adott nyelvi adat helyreállítása nehézségének megfelelően három szintet különítünk el.

Az első #**1** szint a betűnkénti egyértelmű átírást jelenti: 'Вилмош' → Vilmos[2], mikor az orosz betűket magyar megfelelőjükre cseréljük. Ez történhet a hivatalos szabályzat (Zoltán, 1981) szerint, ugyanakkor látjuk, hogy ez a módszer a legritkább esetben elegendő.

Említettük, hogy az orosz és magyar betűk sok esetben nem egy az egyben felelnek meg egymásnak. Esetünkben az jelenti a problémát, hogy *egy* orosz betűnek *több* magyar is megfelelhet (2. táblázat). A második #**2** szinten ezt a problémát oldjuk meg lényegében azáltal, hogy végigpróbálgatjuk a lehetséges betűket, hogy értelmes magyar szót kapjunk.

[2] Ha hangsúlyozni akarjuk, hogy egy szövegrész orosz, akkor sima idézőjelekkel, magyar esetén pedig aláhúzással jelöljük.

orosz	→ **magyar**
М**о**ноки	M**o**noki
Микл**о**ш	Mikl**ó**s
К**о**лмар	K**a**lmár
Тегл**о**ш	Tégl**á**s

2. táblázat. Egy orosz betűnek több magyar is megfelelhet, az orosz 'o'-nak például leggyakrabban o, ó, a vagy á.

A fentiekhez adódnak hozzá a 2. részben említett torzulások, azaz amikor adott pozícióban egyáltalán nem a megfelelő betű szerepel, illetve amikor félrehallás, félreolvasás és a szöveg nem értése miatt különféle összetettebb hibák és következetlenségek kerülnek az anyagba (3. táblázat). Ezeket a harmadik #**3** szintű „okos" helyreállítás igyekszik minél jobban megoldani.

orosz	→ **magyar**	**hibafajta**
Баконь**е**омба**н**дхель	Bakonyszombathely	más betű: 'c' helyett 'e' fölösleges: 'н'
Бала**ш**адарма	Balassagyarmat	kimarad: 'т'
Бик**ши**чаба	Békéscsaba	csere: 'иш' helyett 'ши'
Бешене**лект**	Besenyőtelek	más betűcsoport

3. táblázat. Hibafajták.

Míg a Вилмош → Vilmos esetén elegendő az #**1** szint, a Ковач → Kovács megoldásához szükséges a #**2** szint, mivel meg kell állapítani az orosz 'a' aktuális megfelelőjét. Végül a Шаторомоуйгел → Sátoraljaújhely esetén #**3** szint szintű eljárás szükséges a megfelelő helyreállításhoz: a betűmegfeleltetéseken kívül fel kell oldani a 'омо' → alja konverziót is. A 4. táblázatban néhány könnyebb és nehezebb példa látható.

Nagyon valószínű, hogy az első torzulás (ld. 2. rész) során keletkeztek a félrehallásos, elírásjellegű hibák, a második torzulás során pedig a félreolvasásos, OCR-jellegű hibák. Az előbbire példa a Рещелог (4. tábl/1.), mivel a k→g félrehallás is könnyen elképzelhető, a két betű formája viszont eltérő; az utóbbira pedig a Леретц (4. tábl/7.), mert a n→t félrehallás nem valószínű, viszont a megfelelő orosz betűk ('н' és 'т') alakja hasonló. Mindenesetre az anyagban a két hibaosztály szerencsétlen keveredését látjuk.

A 4. táblázat azt is illusztrálja, hogy ennél a feladatnál (és más hasonlóknál is) minden bizonnyal meglesz a táblázatban látható három osztály: a géppel megoldható esetek, a géppel nem, de manuális munkával megoldhatók és a manuális munkával sem megfejthetők. Célunk a második osztály méretének csökkentése

	orosz	#1 szint	#3 szint	ember	gép
1.	Репцелог	Repcelog	Répcelak	✓	✓
2.	Фейняшлидке	Fejnyáslidke	Fényeslitke	✓	✓
3.	Хатоваин	Hatovain	Hatvan	✓	✓
4.	Лайошминш	Lajosmins	Lajosmizse	✓	✓
5.	Яцберин	Jácberin	Jászberény	✓	✗
6.	Ямуш	Jámus	János	✓	✗
7.	Леретц	Leretc	Ferenc/Lőrinc	✓	✗
8.	Блодентмигайн	Blogyentmigájn	→ 11. oldal	?	✗
9.	Аирг	Airg	???	?	✗
10.	Алохупкуя	Alohupkuja	???	?	✗

4. táblázat. Példák hozzávetőleges nehézségi sorrendben. Az első négy példát a jelenlegi rendszerünk jól kezeli. Az 5-7. példák emberi intelligenciával biztosan kitalálhatók, bár előfordul, hogy több megoldás is jónak tűnik.

az első növelése révén, azaz géppel minél jobban megközelíteni az emberi teljesítményt.

5. Feldolgozott adatmezők

A munkálat során azokkal az adatmezőkkel foglalkozunk, amelyeket nem fordítani, hanem helyreállítani kell (vö: 3. rész), azaz ami nem orosz szöveg, hanem magyar szöveg cirill betűkkel.

Ide tartozik: (1) vezetéknév; (2) keresztnév; (3) apai keresztnév; (4) születés helye; (5) fogságba esés helye. A nálunk megszokott „anyja neve" helyett a szovjet hatóságok – az orosz nevek felépítésének megfelelően – az apai keresztnevet jegyezték fel. A két hely mező ország, megye, járás, település, utca, házszám részekből, illetve ezek épp jelenlévő részhalmazából áll. Az utcával és a házszámmal nem foglalkozunk, főképp azért, mert nem áll rendelkezésre az utcaneveket tartalmazó átfogó lista.

A fordítást igénylő mezők tehát kimaradnak: dátumok, fogadó tábor, rendfokozat, amennyiben elhunyt hol nyugszik, elbocsátó tábor, megjegyzés, azonosítók.

6. A helyreállítás módszere

A bevezetésben említett cirill betűs adatbázisból indultunk ki, a szkennelt kartonokkal nem foglalkoztunk. Utóbbi egy teljesen más léptékű feladat lenne, amit az orosz partner manuális munkával lehetőségeihez képest megfelelően elvégzett. Rendszerünkben azt a megközelítést választottuk, hogy az eredeti cirill verzióból származtatunk több lehetséges magyar átírást, Morse (2005) éppen az ellenkező irányt választotta.

6.1. Előfeldolgozás

Három részfeladatot végzünk el az előfeldolgozás keretében: a női keresztneveket és az orosz formában megjelenő apai neveket speciálisan kezeljük, valamint a helyadatokat releváns adatmezőkre bontjuk.

A foglyok között kb. 1%-ban fordulnak elő nők. Ha keresztnévként férfi és női neveket is elfogadunk, akkor számos olyan hiba adódik, hogy férfi nevet női-ként kezelünk: Пауль → Paula (helyesen: Paul/Pál); Матия → Maja (helyesen: Matija/Mátyás); Алоис → Aliz (helyesen: Alois/Alajos). Ezért azt a megoldást választottuk, hogy a keresztneveknél csak férfi neveket fogadunk el, a női neveket pedig egyedileg kezeljük, azaz listába gyűjtve egyenként állítjuk helyre.

Az apai keresztnév mezőben sokszor megjelenik az oroszra jellemző -вич/-вна végződés, akár egyértelműen magyar névhez illesztve is: Чилик Юзеф Имре**вич** → Csilik József Imre**vics**. Ezt a végződést elhagyjuk.

A hely mezőket (születés helye és fogságba esés helye) valódi adatmezőkre bontjuk: 1. ország, 2. megye, 3. járás, 4. település. A felbontást az országok listája és a meglévő rövidítések alapján végezzük (5. táblázat), a feladat a sorrendi és egyéb következetlenségek miatt okoz nehézséget.

Венгрия, обл. Пешт, д. Вечешь → Magyarország, Pest megye, Vecsés település
с. Сигатуйфалу, обл. Пештмеде, Венгрия → Magyarország, Pest megye, Szigetújfalu település

5. táblázat. Két könnyen felbontható helyleírás. Rövidítések: обл. = область → megye, д. = деревня → falu. Az esetenként megjelenő -меде tagot elhagyjuk.

6.2. Átíró szabályok

Az átírást végző szabályrendszerek létrehozásához manuális munkával megvizsgáltuk az egyes cirill karakterek 100-100 véletlenszerű előfordulását és ez alapján állapítottuk meg a lehetséges magyar megfelelőket. Az átíró szabályok két változatban készültek. A szigorú vagy strict változat egy magyar megfelelőt tartalmaz (pl. 'д' → d), a laza vagy loose pedig többet (pl. 'д' → d|gy|t). Az előbbi az #**1** szintnek felel meg, az utóbbi a #**2** szintnek illetve a #**3** szint nem megfelelő betűkre vonatkozó első felének. Az eredendően betűkre, betűpárokra vonatkozó szabályokat tartalmazó szabályrendszert kiegészítettük az országnevek fordításával, valamint a településnevek végén gyakran előforduló elemek (pl.: *-falva*, *-háza*) helyreállított alakjának listájával.

Minden adatmezőhöz tartozik egy elvárt értékeket (vezetékneveket, keresztneveket, országokat, településeket stb.) tartalmazó gazetteer-lista, egy opcionális gyakorisági („fontossági") lista, valamint egy szabályrendszer a fenti két verzióban. Ezt egyben az adatmezőhöz tartozó *eszközkészlet*nek nevezzük.

6.3. Az algoritmus

A helyreállítást megvalósító algoritmus a következő lépésekből áll.

1. Előkészítjük az adatmezőhöz rendelt eszközkészletet.
2. Átírjuk az adatot a laza átíróval.
3. Az így kapott reguláris kifejezést illesztjük a listára.
4. Megtalálható a listán így? Ha igen, visszaadjuk az összes találatot. ✓
5. Ha nincs, akkor közelítő kereséssel keressük a szigorú átiratot a listán.
6. Megtalálható? Ha igen, visszaadjuk a legjobb találatot. ✓
7. Egyébként: visszaadjuk a szigorú átiratot. ✓

Az algoritmusnak három (pipával jelölt) kimeneti pontja van, ezek rendre megfelelnek az #**1**+#**2** szintnek, a #**3** szintnek illetve annak az esetnek, amikor semmilyen módon nem sikerült a megfelelő lista egy elemeként azonosítani a helyreállított alakot, így a puszta szigorú átiratnál jobbat nem tudunk mondani. 'Имре' → Imre esetén már az #**1** szint eredményt adna. Ha 'Андрош' a kiinduló adat, akkor reguláris kifejezés segítségével találjuk meg a helyes András alakot (vö: 2. táblázat). 'Форенц' bemenetnél a regex nem segít, mert az orosz 'о'-nak a laza átíró nem felelteti meg az e-t. Itt a közelítő keresés találja meg a Ferenc alakot. Végül, ha a 'Момольсильтер' szóalak az adat, akkor egyik megközelítés sem jár sikerrel, így csupán a szigorú átíróval létrehozott Momolszilter alakot tudjuk visszaadni.

A módszert python nyelven implementáltuk, a 5. lépésben a közelítő keresést a `difflib` csomaggal valósítottuk meg. Az 4. lépésben a kapott találati halmaz sorrendezéséhez két szempontot veszünk figyelembe: egyrészt a találatoknak a szigorú átirathoz való hasonlóságát, valamint lehetőség szerint a találatok általános gyakoriságát. Előbbit a `difflib` megfelelő függvényével számítjuk ki, utóbbi adat a vezetéknevek és a keresztnevek esetében állt a rendelkezésünkre egy első világháborús veszteséglista formájában. A találatokat a két adatból képzett közös pontszám sorrendjében, a pontszámmal együtt adjuk vissza. Az 'Андрош' alakból laza átíróval képzett regex – `(A|Á)(n|m)(d|gy|t)(r|l)(a|á|o|e)(s|sch)` – 192 különböző alakot fed le. Erre a lazaságra általában szükség is van, mert az egyes adatelemek nagyon sok változatban valóban előfordulnak.

6.4. Idegennyelvű szövegek

Az eredeti kiindulópontunkkal szemben, miszerint *magyar* nyelvű adatokat kell helyreállítanunk, kitűnt, hogy egyéb célnyelv (vö: 3. rész) is előfordul. Például a német. Egyrészt számos német nemzetiségű hadifogoly is volt, másrészt sokakat akkor fogtak el, mikor a front már Ausztria területén járt, így az elfogás helye osztrák település (6. táblázat).

A német adatelemek kezelésére szükséges volt létrehozni egy komplett *orosz–német* átíró szabályrendszert, amiben a németnek megfelelő szabályok kaptak helyet: 'ц' → c helyett 'ц' → z; 'в' → v helyett 'в' → w; 'ай' → aj helyett 'ай' → ei; 'ой' → oj helyett 'ой' → eu stb.

Гроц	→ Graz
Лииц	→ Linz
Фрайштадт	→ Freistadt
Дойчландберг	→ Deutschlandsberg
Штокаров	→ Stockerau
Цветел	→ Zwettl

6. táblázat. Osztrák településnevek az adatbázisban.

A rendszer működését alkalmassá tettük arra, hogy bizonyos feltétel teljesülése esetén alternatív eszközkészlet használatára lehessen váltani. Azon helyek esetén tehát, melyekben szerepel az 'Австрия' alak, átváltunk az orosz-német szabályok + osztrák településlista eszközkészletre.

7. Eredmények, megfontolások

7.1. Kiértékelés

Az előzőekben bemutatott eljárás kiértékelésére két mértéket használtunk. A *megalapozott helyreállítások aránya (M)* egy fedés jellegű mérték, azt mutatja meg, hogy az adatok hány százalékára tud a módszerünk a buta szigorú átírásnál jobb megoldást adni (ld. az algoritmus 4. és 6. pontja a 6.3. részben). A *helyes helyreállítások aránya (H)* egy pontosság jellegű mérték, azt mutatja meg, hogy az adatok hány százalékára van valóban helyes helyreállítás. A nagyon specifikus feladat miatt más módszerekkel való összevetésre nincs lehetőség. Az M és H értékeket a 7. táblázat mutatja be. Az adatok kezelhetőség szempontjából az M érték alapján kétfelé oszlanak: a nevek és az országok (✓a táblázatban) jól kezelhetők (M = 95-100%), a többi helyadat sokkal nehezebb (M = 50-70%). Utóbbiban valószínűleg közrejátszanak a felbontás nehézségei (vö: 6.1. rész).

A H értékeket is figyelembe véve négy csoport látszik. A keresztnevek és az országnevek (2-4. és 8. sor) szinte mindegyikére van ajánlata az algoritmusnak és lényegében az összes ajánlat helyes is. A vezetékneveknél (1. sor) a helyesség alacsonyabb. Közrejátszik, hogy keresztnevekhez képest sokkal (∼50x) több vezetéknév van, jóval több nehezen megfejthető példa fordul elő: 'Хумато' → Homoga?, 'Турuел' → Turul?; valamint, hogy a *B. Kovács* típusú összetett vezetéknevek nincsenek jelenleg kezelve. A megyéknél (5. sor) kevesebb helyen tud tippet adni az algoritmus, olyankor viszont szinte mindig helyes a tipp. Itt jellemző hiba, hogy keverednek a különböző méretű közigazgatási egységek: *Derecske* vagy *Gyömrő* például megjelenik megyeként is az adatbázisban. A település és a járás (6-7. és 10-11. sor) a legnehezebb. A járás kisebb fontosságú és eléggé ritkán is fordul elő, a település viszont kiemelten fontos a hadifoglyok azonosításhoz. Itt sajnos az M és a H is alacsony, leginkább ezen szükséges javítani a jövőben. A nehezen megfejthető példák (pl.: Фоло, Улалануш) mellett gondot jelentenek itt a nem-osztrák külföldi települések, valamint a gyakran előforduló hosszú te-

	adatmező	M	helyes	/ összes	= H	H/M
✓ 1.	vezetéknév	95,8%	76	100	76%	79%
✓ 2.	keresztnév	95,0%	92	100	92%	97%
✓ 3.	apai keresztnév	95,2%	70	78	90%	95%
✓ 4.	születés: ország	99,9%	45	45	100%	100%
5.	születés: megye	70,7%	32	49	65%	92%
6.	születés: járás	60,9%	7	15	47%	77%
7.	születés: település	65,9%	31	61	51%	77%
✓ 8.	fogságba esés: ország	99,9%	33	33	100%	100%
9.	fogságba esés: megye	67,7%	11	12	92%	—
10.	fogságba esés: járás	46,2%	1	4	25%	54%
11.	fogságba esés: település	67,5%	29	56	52%	77%
	összesen	**85,5%**	**427**	**553**	**77%**	**90%**

7. táblázat. Eredmény: az adatok 77%-ához tudunk helyes helyreállított alakot rendelni. A kiértékelésben a megalapozott helyreállítások aránya (M, „fedés”) és a helyes helyreállítások aránya (H, „pontosság”) szerepel. Előbbit a teljes, 682000 rekordot tartalmazó adatbázis alapján számoltuk, utóbbit az adatbázisból képzett 100 rekordból álló random mintán állapítottuk meg manuális kiértékeléssel. Az *összes* mező mutatja, hogy 100 sorból hányban volt jelen a szóban forgó adat. A H/M arány arról informál, hogy a megalapozott helyreállítások hány százaléka helyes valóban. A 9. sorban – vélhetően a H-hoz használt minta kis mérete miatt – nem értelmezhető érték adódik.

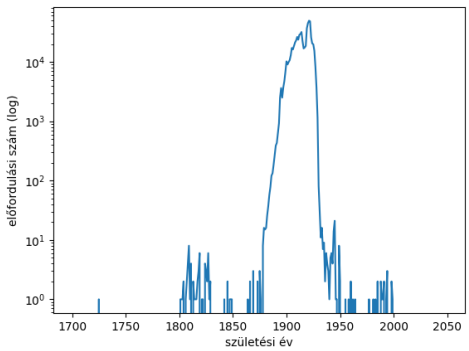
lepülésnevek, melyek gyakran számos hibát tartalmaznak (pl.: ’Яскорогенуй’ → Jászkarajenő?, ’Пишпекляний’ → Püspökladány?).

Látjuk, hogy az M és H értékek sok helyen összecsengenek: ahol tud valamit mondani az algoritmus, ott legtöbb esetben helyes is a javasolt helyreállítás. Az M-kiértékelés automatikus (össze kell számolni), a H-kiértékelés manuális munkát igényel. A H/M arány megmutatja, hogy az algoritmus által szolgáltatott megoldások mennyire jók. Ha ez magas, annak az az előnye, hogy megspórolhatjuk a munkaigényes H-kiértékelést, mert ekkor a H érték jól becsülhető az M értékkel. Ez az eset azokra az adatmezőkre jellemző, ahol lehetséges helyes adatok száma alacsony.

7.2. A szabadszöveges adatbázisokról

A kiinduló adatbázisunk egy *kézzel készült, korlátozatlan, szabadszöveges* adatbázis. Ez azt jelenti, hogy az adatbevivő nincs semmilyen értelemben korlátozva – például legördülő menüből való választás vagy típusellenőrzés révén –, azaz teljesen szabadon azt ír be az adatmezőkbe, amit csak akar. Az ilyen adatbázisok szükségszerűen következetlenek, mivel nincs olyan mechanizmus, ami biztosítaná az adatok egységességét: hogy ugyanazt mindig ugyanúgy jelöljük, az eltérő dolgokat pedig mindig eltérően.

Amellett, hogy az ilyen adatbázisokban egy adat több validnak mondható formában fordul elő, az ilyen adatbázisokba ellenőrzés híján számos hiba, elírás is belekerül. Azt látjuk, hogy ha nem legördülő menüből kell választani, akkor még a születési év adatot is el lehet rontani (2. ábra). A tanulság az, hogy az adatbázisok készítésekor szükséges az ellenőrzés, az egységesítő mechanizmus.



2. ábra: A adatbázisban szereplő *születési év* adatmező értékeinek eloszlása. A második világháborús hadifoglyok adatai között előfordul 1725-ös és 1999-es születési év is.

Viszont van olyan eset is, amikor valóban szabad kezet akarunk adni az adatszolgáltatónak/adatrögzítőnek. Véleményünk szerint ilyen eset a közvéleménykutatás. Azt gondoljuk, hogy ha egy közvéleménykutatási kérdés esetén – főként ha *miért*-es kérdésről van szó – a válaszadónak néhány előre megadott választási lehetőség közül kell választania, akkor a kutatás szükségszerűen veszít a hitelességéből ahhoz képest, ha a válaszokat szabadon fogalmazhatja meg, például mivel adott esetben véleményét jól visszaadó válasz egyszerűen nem szerepel a lehetőségek között.

A korlátozott módon készülő adatbázisokat persze sokkal könnyebb kiértékelni. Ugyanakkor a korlátozatlan adatbázisok adatainak értelmezése is megvalósítható: nyelvtechnológiai eszközökkel. Két esetben kaphat tehát szerepet a nyelvtechnológia: amikor nem történt előzetes adatellenőrzés/korlátozás (pl. a jelen tanulmányban tárgyalt hadifogoly-adatbázis) illetve amikor nem akarunk előzetes adatellenőrzést/korlátozást (pl. közvéleménykutatás).

A szabadszöveges adatbázisok értelmezési-feldolgozási munkálatait érdemes három szakaszra bontani: (1) adatvizsgálaton alapuló szakasz; (2) gyakorisági hibaelemzésen alapuló szakasz; (3) manuális szakasz. Az első szakaszban valamilyen automatikus rendszer áll elő, ami az adatok jelentős részét kezelni képes, a második szakaszban ezt finomítjuk a felfedett gyakori hibák javítása révén. Tudva azt, hogy ha a tökéleteshez közeli eredményt szeretnénk, akkor nem le-

het megspórolni a manuális szakaszt, a második szakaszban azokkal a hibákkal foglalkozunk, amelyek javítása a legnagyobb haszonnal jár.

A jelen tanulmányban feldolgozott adatbázis a többrétű torzulás miatt a szabadszöveges adatbázisoknak is a szélsőségesen következetlen és sokféle hibával teli fajtájához tartozik. Kezeléséhez a fent (6. rész) ismertetett szabályalapú megközelítést alkalmaztuk. Azért fogtunk hozzá így, mert egyrészt egy teljesen egyedi feladat konkrét problémáit kellett megoldani behatárolt méretű adathalmazon, valamint tanulóadat híján a gépi tanulási módszerek alkalmazására nem volt lehetőség. Ilyenkor ma is lehet létjogosultsága a szabályalapú módszereknek.

7.3. Példák

A 8. táblázatban egy engedéllyel közzétett valódi teljes példa látható.

vezetéknév	**Галь**	**Gál**
keresztnév	**Тибор**	**Tibor**
apai keresztnév	**Эмиль**	**Emil**
születési év	1915	1915
születés helye	**г. Сольнок**	**Szolnok település**,
	ул. Санопи, 17	Szanopi (?) utca 17.
fogságba esés helye	**г. Цветел, Австрия**	**Ausztria, Zwetel település**
nemzetiség	венгр	magyar
fogságba esés ideje	12.05.1945	12.05.1945
elbocsátás ideje	08.07.1947	08.07.1947
fogadó tábor	сдан лагерь № 36	36-os tábor
rendfokozat	лейтенант	hadnagy
elbocsátó tábor	лагерь № 313	313-as tábor

8. táblázat. A helyreállító rendszer által kezelt adatok vastagítva láthatók. Egy helyen nem tökéletes a megoldás: a *Zwetel* helyesen *Zwettl* lenne.

Annak illusztrálására, hogy valóban előfordulhattak félreolvasási hibák (vö: 4. oldal) a kartonok elektronikus rögzítésekor, bemutatunk egy eredeti kartont (3. ábra).

A végső manuális szakaszra maradó adatok helyreállításának nehézségét két példán mutatjuk be. Ilyenkor előfordul, hogy egy-egy adat megfejtése önmagában kutatómunkát illetve több kutató együttműködését igényli. A 4. táblázat 8. bejegyzéseként látható Блодентмигайн helyreállítva Bűdszentmihály. Itt kezelni kell a szó végén lévő ’йн’ variációt, a kieső sz-t, valamint meg kell fejteni, hogy hogyan változhatott az ű az orosz ’ло’ betűkapcsolattá. Ez az eset mindkét torzulástípust példázza, ugyanis minden valószínűség szerint az adatrögzítéskor lett ű-ből ’ю’ megfelelőbb orosz betű híján; majd a digitalizáláskor ’ю’-ból ’ло’ félreolvasás révén. A 6.3. részben idézett Момольсильтер helyreállítva Mosonszentpéter. Itt az segített, hogy az adott napon Mosonszentpéteren esett fogságba

3. ábra: Egy eredeti karton. Az írás értelmezése nagy gyakorlatot igényel.

a hadifoglyok nagy része, és az adatbázisban szerepeltek az idézett orosz alakra sokban hasonlító, de könnyebben megfejthető verziók is.

Köszönjük Nyéki Bence, Orosz Ferenc, Beke Gábor és Szatucsek Zoltán közreműködését a munkálatokban, illetve hozzájárulásukat a fenti példák megoldásához.

8. Elérhetőség

A helyreállító rendszer részletes technikai információkkal, teljes szabályrendszerekkel, listákkal, futtatható programmal és minden egyébbel elérhető a `https://github.com/dlt-rilmta/hadifogoly-adatbazis` címen. Az eredeti adatokat ez a repozitórium nem tartalmazza, rendelkezésre áll viszont egy adatmezőnként külön-külön randomizált adatfájl (`data/pseudo_1000_42.csv`), amin az eljárás a `make transcribe FILE=pseudo_1000_42` paranccsal futtatható.

9. Továbbfejlesztési lehetőségek

Az ismertetett módszer nem oldja meg maradéktalanul a kitűzött feladatot. Konkrét feladat lévén az elvi cél a teljes, 100%-os megoldás, ehhez, ahogy említettük, mindenképpen szükséges egy manuális munkaszakasz is. Ugyanakkor a teljesítmény növelése érdekében számos továbbfejlesztési lehetőség adódik.

A településlisták jelenleg magyar és osztrák településeket tartalmaznak. Sokszor előfordulnak hiányzó települések. A „magyar verziókat” (Bécs, Bukarest stb.) a magyar (átírószabályokhoz tartozó) listához szükséges hozzátenni, az idegennyelvűeket külön listákba szükséges gyűjteni, és a 6.1. rész elején említettek miatt nyelvenként kezelni. Ehhez minden nyelvhez új átíró szabályrendszer szükséges, valamint egy módszer az aktuális nyelv meghatározására.

Megfontolható, hogy hogyan lehetne automatizálni a fogságba esés idejére alapuló a 7.3. rész legvégén említett ötletet a településnevek jobb azonosítására.

Érdemes lehet újból felmérni a hagyományos vagy neurális gépi tanulás alkalmazhatóságát. Manuális úton természetesen lehet tanítóadatot gyártani, ezzel eddig nem foglalkoztunk. A kiinduló adat ismertetett extrém tulajdonságai miatt azonban nem vagyunk meggyőződve arról, hogy a gépi megközelítés esetünkben jelentősen jobb eredményt adna.

10. Összefoglalás

Tanulmányunkban egy orosz-magyar átíró és helyreállító rendszert mutattunk be, melynek célja a magyar hadifoglyok adatbázisából kiindulva értelmezni és helyreállítani az eredeti magyar adatokat a cirill betűs forma alapján. A kidolgozott szabályalapú módszer az adatok 77%-ához tud helyes helyreállított alakot szolgáltatni. Ez a kiinduló adatbázis minőségét tekintve megfelelő eredmény. A teljesítmény növelésére elsősorban a helyadatok feldolgozásában nyílik lehetőség a jövőben.

Az adatbázis jól példázza a kézzel készült, korlátozatlan, szabadszöveges adatbázisok szükségszerű következetlenségét; az automatikus feldolgozást esetünkben még az adatok kétszeres torzulása is nehezíti. Az ilyen adatbázisok feldolgozását éppen a nyelvtechnológiai eszközök használata tudja megfelelően támogatni.

Hivatkozások

Bradley, J.: The Mari web project's orthography helper(s) (2020), https://www.univie.ac.at/maridict/site-2014/transcription-general.php

Haader, L.: Az ómagyar kódexek hibatipológiájának kutatásáról. In: Korompay, K., Stemler, Á., Terbe, E., C. Vladár, Z. (szerk.) Forráskutatás, forráskiadás, tudománytörténet II., pp. 23–33. Magyar Nyelvtudományi Társaság (2014)

Katona, P., Szikla, G.: A Magyar Nemzeti Levéltár Hajdú-Bihar Megyei Levéltár adatbázisai. Új Nézőpont 1, 61–81 (2014)

Morse, S.P.: Searching the Gulag database in one step (2005), https://stevemorse.org/russian/gulag.html

Prószéky, G., Naszódi, M., Kis, B.: Recognition assistance - treating errors in texts acquired from various recognition processes. In: COLING 2002: The 17th International Conference on Computational Linguistics: Project Notes. pp. 1263–1267. Taipei, Tajvan (2002)

Zoltán, A.: A cirillbetűs írású szláv nyelvek szavainak és neveinek magyar helyesírása. MTA I. Osztály Közleményei 32, 171–192 (1981)