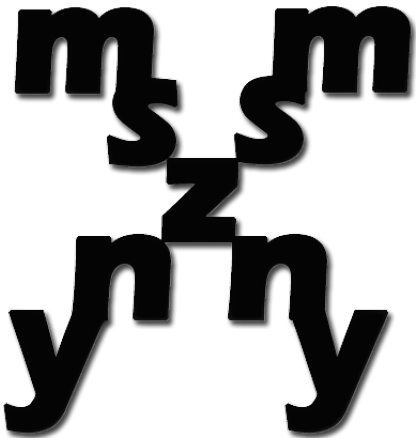
XX. Magyar Számítógépes Nyelvészeti Konferencia



Szerkesztette:
Berend Gábor
Gosztolya Gábor
Vincze Veronika

Szeged, 2024. január 25–26.

Szerkesztette:
Berend Gábor, Gosztolya Gábor, Vincze Veronika
{berendg,ggabor,vinczev}@inf.u-szeged.hu

Felelős kiadó:
Szegedi Tudományegyetem
TTIK, Informatikai Intézet
6720 Szeged, Árpád tér 2.

ISBN: 978-963-306-973-8

Szeged, 2024. január

Az MSZNY 2024 konferencia szervezője:
HUN-REN–SZTE Mesterséges Intelligencia Kutatócsoport

Előszó

2024. január 25–26-án immáron huszadik alkalommal kerül sor a Magyar Számítógépes Nyelvészeti Konferencia megrendezésére. A konferencia fő célkitűzése a kezdetek óta állandó: lehetőséget biztosítani a nyelv- és beszédtechnológia területén végzett kutatások eredményeinek ismertetésére és megvitatására, ezen felül pedig a különféle hallgatói projektek, illetve ipari alkalmazások bemutatására.

Az idei évben a 24 beküldött cikkből gondos mérlegelést követően 19 cikk került elfogadásra, melyek témája a nyelv- és beszédtechnológia számos szakterületét lefedi a legújabb nyelvi modellek bemutatásától kezdve a beszédtechnológia eredményein keresztül egészen a gépi fordításig.

Nagy örömet jelent számunkra, hogy Sebők Miklós elfogadta meghívásunkat, aki plenáris előadását *Nagy nyelvi modellek az összehasonlító politikatudományban: Közpolitikai témák klasszifikációja a Babel-rendszerrel* címmel fogja megtartani.

Az idei évben is különdíjjal jutalmazzuk a konferencia legjobb cikkét, mely a legjelentősebb eredményekkel járul hozzá a magyarországi nyelv- és beszédtechnológiai kutatásokhoz. Ezen felül immár hatodik alkalommal osztjuk ki a legjobb bíráló díját, amellyel a bírálók fáradságos, ugyanakkor nélkülözhetetlen munkáját kívánjuk elismerni.

A szervezőbizottság nevében,
Ács Judit,
Berend Gábor,
Gosztolya Gábor,
Ligeti-Nagy Noémi,
Nemeskey Dávid Márk,
Novák Attila,
Simon Eszter,
Sztahó Dávid,
Vincze Veronika

Tartalomjegyzék

Bírósági határozatok automatikus mondatszegmentálásának hatékonyságmérése

Csányi Gergely Márk[1], Lakatos Dorina[1,2], Üveges István[1,3], Vági Renátó[1,4], Megyeri Andrea[5], Fülöp Anna[5], Nagy Dániel[1], Vadász János Pál[1,6]

[1]MONTANA Tudásmenedzsment Kft.
[2]HUN-REN Számítástechnikai és Automatizálási Kutatóintézet
[3]HUN-REN Társadalomtudományi Kutatóközpont
[4]Eötvös Loránd Tudományegyetem, Állam- és Jogtudományi Doktori Iskola
[5]Wolters Kluwer Hungary Kft.
[6]Nemzeti Közszolgálati Egyetem, Információs Társadalom Kutatóintézet
{csanyi.gergely,lakatos.dorina,uveges.istvan,
vagi.renato,nagy.daniel,vadasz.pal}@montana.hu
{andrea.megyeri,anna.fulop}@wolterskluwer.com

Kivonat A természetesnyelv-feldolgozási feladatok gyakori építőeleme a mondatokra történő szegmentálás, amely azonban a jogi szövegek esetében hagyományosan problémás terület. Jelen cikkünkben bemutatjuk a bírósági határozatok szegmentálására „hangolt“ szabályalapú eszközünket, összemérve azt pontosságban és futásidőben más magyar nyelvre alkalmazható mondatszegmentálókkal az eddigiekben mondatszegmentálás mérésére használt Szeged Treebank és UD korpuszokon, valamint egy csak magyar nyelvű bírósági határozatokat tartalmazó korpuszon. A szegmentálónk a Szeged Treebank korpuszon összességében a Stanza szegmentálóhoz hasonló eredményt ért el, azonban a jogi alkorpuszon már a legjobb modell volt, amely nem látta korábban ezt az adatot. A bírósági határozatokon a legjobbnak bizonyult a megközelítésünk. Megvizsgáltuk a tördelés ismeretének hatását a szegmentálókra, valamint a szövegek doménjeinek hatását a szegmentálókra.
Kulcsszavak: mondatra bontás, mondatszegmentálás, bírósági határozatok

1. Bevezetés

A mondatszegmentálás egy olyan alapvető művelet, amely sok természetesnyelv-feldolgozó projekt vagy alkalmazás alappillére. Előfordulhat például, hogy mondatszintű adatokra van szükségünk egy extraktív (Parikh és mtsai, 2021; Csányi és mtsai, 2023) vagy absztraktív (Yang és mtsai, 2021) kivonatolás elvégzéséhez, gépi fordítás során (Laki és Yang, 2022), vagy egy szöveg mondatait szeretnénk külön klasszifikálni (Ghosh és Wyner, 2019; Santosh és mtsai, 2023), esetleg hivatalos vagy jogi szövegek (mondatonkénti) közérthetőségi szempontú osztályozása esetében (Üveges, 2022).

Magyar nyelvre többféle mondatszegmentáló megoldás is ismert, pl. a szabályalapú quntoken (Mittelholtz, 2017), az LSTM alapú Stanza (Qi és mtsai, 2020) illetve a HuSpaCy konvolúciós háló és transzformer alapú megoldásai (Orosz és mtsai, 2022, 2023). Ezek egyenként különböző előnyökkel és hátrányokkal bírnak: van amelyik pontosabb de lassabb, van olyan ami kevésbé pontos de gyors. Egy projekt során a pontos működés és a sebesség melletti szempont, hogy milyen erőfeszítés mellett lehetséges a szegmentáló hangolása egy adott feladatra, szövegtípusra, hiszen előfordulhatnak olyan jellegzetességek az újabb adatban, amelyeket a szegmentálók alapértelmezetten nem képesek hibamentesen kezelni. A jogi szövegek, ezen belül a bírósági határozatok több ilyen tulajdonsággal is bírnak, pl. rengeteg a pontot tartalmazó, de mondatvéget nem jelölő szövegrész (ügyszámok, hivatkozások stb.), anonimizálásból fakadó problémák (pl. három pont az anonimizált szövegrész helyett, monogramok), hosszú felsorolások, címek a szövegben stb.

A magyar nyelvű mondatszegmentálók komparatív mérésére több korpusz is használatos, ilyenek pl. a magyar Universal Dependency korpusz (De Marneffe és mtsai, 2021), amely tulajdonképpen a Szeged Treebank egy részhalmaza (Csendes és mtsai, 2005), amely szintén használatos összehasonlító mérésekhez (Váradi és mtsai, 2018). A Szeged Treebank több doménből tartalmaz szövegeket: rövidhírek, iskolai fogalmazások, jogszabályok szövegei, újságcikkek, valamint szépirodalmi és számítástechnikai szövegek. Terjedelme kb. 200 000 token minden domén esetében. Jelen munkánk során a Szeged Treebank 2.0-ás verzióját használtuk fel. A cikkben bemutatjuk a bírósági határozatok mondatszegmentálására továbbfejlesztett szabályalapú megoldásunkat, összemérve a fentebb említett korpuszokon a magyar nyelvre használatos mondatszegmentáló eszközökkel.

2. Korpusz bírósági határozatokból

Szerettük volna a mondatszegmentálónkat magyar bírósági határozatokon is összehasonlítani a többi mondatszegmentálóval, ezért készítettünk egy erre alkalmas korpuszt, ennek az elkészítési folyamatát mutatjuk be. A korpusz szabadon nem elérhető. A munkánk során az egyszerűség kedvéért a továbbiakban a korpuszra HuCoDe (**Hu**ngarian **Co**urt **De**cisions) néven hivatkozunk. A magyar bírósági határozatok hat főbb jogterületről származnak: büntető, gazdasági, katonai büntető, közigazgatási, munkaügyi és polgári. A mintakorpusz készítéséhez összesen 190 710 darab bírósági határozat volt elérhető, az alábbi jogterület szerinti eloszlással. A tokenek számát whitespace-ek mentén felbontva és az üres tokeneket kiszűrve számoltuk, a dokumentum méretek összehasonlíthatósága céljából.

Az 1. táblázatban bemutatott korpusz részhalmazát képeztük oly módon, hogy az egyes jogterületekből kb. 50 ezer tokennyi adatot választottunk ki, egybevéve a katonai büntető és büntető jogterületeket, de a két jogterület viszonylatában nagyjából megtartva az eredeti korpuszban mért egymáshoz viszonyított arányt, és nem választottunk dokumentumot az egyéb kategóriából. Mivel a do-

1. táblázat. Dokumentum számosságának eloszlása jogterületenként

Jogterület	**Büntető**	**Gazdasági**	**Katonai büntető**	**Közigazgatási**	**Munkaügyi**	**Polgári**	**Egyéb**
Számosság [db]	30 181	23 665	2 866	38 504	16 855	78 611	28
Részarány [%]	15,83	12,41	1,50	20,19	8,84	41,22	0,01
Átl. tokenszám	5 416	3 976	2 792	2 601	2 521	3 078	1 126

kumentumok hossza egyaránt lehet nagyon rövid és hosszú is, a kiválasztáshoz felhasznált dokumentumokat előszűrtük oly módon, hogy azok legalább 1000 tokent tartalmazzanak, felső határnak pedig az adott jogterület átlag tokenszámának 1000 tokenhez viszonyított tükörképét használtuk. Például a büntető jogterület esetében az átlag $\mu = 5416$ token volt, így a felső határ $2 \cdot \mu - 1000 = 9831$ token. A dokumentumokat véletlenszerűen válogattuk jogterületenként a leszűrt halmazból mindaddig, amíg a kiválogatott dokumentumok összes tokenszáma el nem érte az 50000 ± 1000-et. A határozatokból töröltük a keltezést és a bírói aláírásokat tartalmazó szövegrészt a dokumentumok végéről, valamint a fejléc azon részeit amelyek táblázatszerű formátumban tartalmazták a felek és jogi képviselőik adatait, a per tárgyát, előzményügyek azonosítóit stb. Ahol ezek a részek mondatként szerepeltek, ott meghagytuk azokat. A kiválogatott dokumentumok számosságát és tokenszámát a szűrést megelőzően és a szűrést követően a 2. táblázatban mutatjuk be.

2. táblázat. Dokumentum és tokenszámok eloszlása jogterületenként

Jogterület	**Büntető**	**Gazdasági**	**Közigazgatási**	**Munkaügyi**	**Polgári**	**Összesen**
Számosság [db]	21	18	22	23	21	105
Eredeti tokenszám	49 755	49 216	49 881	49 706	49 108	247 666
Szűrt tokenszám	49 258	48 628	48 627	48 324	48 381	243 218

Az elkészített korpusz több szempontból is más a Szeged UD és Szeged Treebank 2.0 korpuszokhoz képest. Ezen korpuszok szövegei eredeti tördelésben nem voltak fellelhetők, illetve azok felépítése alapvetően más, mint akár a doménben legközelebb eső Szeged Treebank jogi részkorpusza. Ez utóbbi jogszabály szövegeket tartalmaz, amely alapvetően más szerkesztettségű, mint egy bírósági határozat, hiszen pl. előbbiek esetében gyakori, hogy egy paragrafusra történő hivatkozással kezdődik egy szöveg, az utóbbiak esetében ez nagyon ritka. Előbbieknél gondosan átnézett, és helyesírási hibáktól mentes szövegről beszélhetünk, szemben az utóbbiakkal, illetve a jogszabályoknál ritkábbak a címek, a fejlécezés hiányzik míg a bírósági határozatok esetében ezek gyakoriak. További jelentős különbség még az anonimizálás jelenléte a bírósági határozatokban, szemben a jogszabályokkal. Ez a gyakorlatban olyan szövegelemek megjelenését jelenti, mint a rengeteg pont egymás után, monogramok és pszeudonímák (pl. Tanú1, helység neve, YYYY stb.).

Mindezekre tekintettel a kézi annotálás során a következő elveket követtük:

– a fejlécben a bíróság nevét és az ügyszámot külön mondatnak tekintettük,

– hasonlóképpen a fejezetcímeket, valamint a felsorolások elemeit,
– kitöröltük a keltezést és a bírói aláírásokat tartalmazó részt,
– kitöröltük a fejlécből azokat a részeket, ahol táblázatos formátumban voltak jelen adatok.

A mondatszegmentálást hét annotátor végezte, mindegyikük előfeldolgozott adatokat kapott kézhez, amelyeket a `montana-splitter` segítségével készítettünk elő, egy egyszerű szövegfájlban minden sorban új mondatot szerepeltetve. A munka során a kérdéses esetek egységes kezelését a vezető annotátor segítette elő.

3. A mintakorpuszok

A Szeged Treebank és UD mintakorpuszokat nem sikerült az eredeti tördelésük szerint megtalálni, vagyis olyan formátumban, ami alapján a szövegeket eredetileg tagoló sortörések, bekezdések stb. rekonstruálhatók lettek volna. Ezért a felbontandó korpuszokat a következő módokon képeztük. Az UD korpusz esetében a CONLLU formátumban felbontott adatok mellett kommentként szerepeltek a felbontás előtti mondatok. Ezeket kigyűjtve és szóközökkel konkatenálva képeztük a felbontandó korpuszt.

A Szeged Treebank esetén a szövegeket a korpusz egy régebbi (2.0 -ás) változatából nyertük ki. Ennek előnye, hogy a szövegek itt TEI (P4 DTD) XML sémában vannak kódolva, vagyis az egyes mondatok külön-külön egy lépésben visszanyerhetők (az `<s>` tag-ek értékeiből). Az így kigyűjtött mondatokat ezután szintén szóközzel konkatenáltuk, az egyes részkorpuszok (pl. jogi szövegek, szépirodalom stb.) szövegeit pedig összeillesztettük a részkorpusz elemeiből.
A felhasznált korpuszok fontosabb alap statisztikáit a 3. táblázat foglalja össze.

3. táblázat. Mondat- és tokenszámosságok

Adathalmaz	Mondatok száma [db]	Tokenek száma [db]
hu_szeged-ud-test	449	10 456
fogalmazások	24 720	338 260
jogi szövegek	9 278	259 960
rövidhírek	9 574	223 549
számítástechnikai szövegek	9 627	209 750
szépirodalom	18 558	233 256
újságcikkek	10 210	219 641
Szeged Treebank	81 967	1 484 416
büntető	1 899	49 258
gazdasági	1 925	48 324
közigazgatási	1 705	48 628
munkaügyi	1 819	48 627
polgári	1 965	48 381
HuCoDe	9 313	243 218

4. A szegmentáló

Az általunk bemutatott mondatszegmentáló megoldás a `sentence-splitter`[1] heurisztikus algoritmust használó szegmentálón alapszik, annak egy bírósági határozatokra továbbfejlesztett verziója, a továbbiakban `montana-splitter`ként[2] hivatkozunk rá. Ez a szegmentáló több nyelvet is támogat, szabály alapon működik, de működése befolyásolható egy segédlista használatával, amely alapértelmezettként 24 nyelvre érhető el, köztük magyarra is. Ebben a listában olyan tokenek definiálhatók, amelyek jellemzően nem szerepelnek egy mondat utolsó tagjaként. A bírósági határozatok szegmentálása során azt tapasztaltuk, hogy az eredeti megoldás jellemzően túlszegmentált, tehát olyan helyeken talált mondatvégeket, amelyek valójában nem voltak azok. Elvégezve a hibaanalízist a következő jellemző hibákat találtuk: ügyszámok, anonimizálásból fakadó problémák (pl. három pont az anonimizált szövegrész helyett, monogramok egymás utáni sokasága, kisbetűs pszeudonímával való mondatkezdés), jogszabályok rövidítései és az ítélkezési gyakorlatra történő hivatkozások okozták a leggyakrabban a problémákat. Pár példa a fentiekre:

- ügyszám: II. Pfv.35.125/2010/4
- ítélkezési gyakorlat: BH.2010.21, KGD.2013.2345 stb.
- három pont: „A ... a keresetében hivatkozott arra, hogy“
- monogramok: „... állítását H.A. és B.L. szemtanúk is megerősítették.“
- pszeudonimizálás: „tanú1 a vallomásában azt állította“
- jogszabály-hivatkozás: „A bíróság perköltséget a 32/2008. (VII. 19.) IM rendelet alapján határozta meg.“

Ezeket a hibákat részben a segédlista bővítésével, részben pedig reguláris kifejezés alapú utófeldolgozási lépésekkel javítottuk ki.

5. Eredmények

A bírósági határozatokon a felbontók kézi kiértékelése során több esetben olyan felbontásokkal találkoztunk, amelyek egyikét vagy másikát is el tudtuk fogadni helyesnek, így az eredmények kiszámítását megelőzően egységesítettük a megoldásokat, kézzel módosítva az egyes szegmentálók által visszaadott eredményeket. Ilyen esetek voltak:

- ahol a fejezet sorszáma (jellemzően római szám) bekerült a mondat elejére - elfogadtuk,
- a bekezdések számozása (pl. [1]), ha külön mondatként lett felbontva, azt elfogadtuk, kivéve, ha a sorszám is több mondatra lett bontva, pl. `"["` és `"1]"` külön mondat,
- számokkal való felsorolásnál, ha a szám külön mondatként lett feltüntetve, azt elfogadtuk,

[1] https://github.com/mediacloud/sentence-splitter

[2] A megoldás kipróbálásához vegye föl a kapcsolatot a szerzőkkel.

– felsorolás elemeit, ha egy mondatba vette és ez nyelvtanilag helyes volt - elfogadtuk,
– ha üres sort adott vissza az adott szegmentáló mondatként, azt töröltük.

Hangsúlyozandó, hogy a tapasztalatok szerint ezek az engedmények nem a saját megoldásunk, hanem a többi szegmentáló számára jelentettek jelentős könnyítést. Például a `huspacy_trf` esetében szinte minden számozott bekezdés esetén a számozás külön mondatként lett szegmentálva, a `stanza` esetében a bekezdések kb. 60%-a volt ebben a jelenségben érintett, a `huspacy_lg` esetén kb. 40%-a, a `huspacy_md` esetben pedig mindössze néhány esetben történt ilyen. A probléma összesen 907 esetben jelentkezett. Az üres sorok mondatként történő felismerésére a `huspacy_md` modell volt hajlamos, összesen 134 üres sort töröltünk ki.

A HuSpaCy modellek esetében nem tudtuk az egész adatot egyszerre feldolgozni, részben memórialimit miatt, részben amiatt, hogy ezeknek a modelleknek van egy maximális méretű bemenetük amelyet a Szeged Treebank adatai túllépnek. Az egyszerre beadott szöveg hosszúságát 200 ezer karakterben maximalizáltuk, úgy hogy a szöveg vége mindig egy mondat vége is volt egyben.

5.1. Fedés és F_1

A szegmentálás kiértékelése némiképp eltér pl. egy hagyományos bináris osztályozó pontosság, fedés, F-mérték számításától, hiszen a szegmentálás helyén elkövetett hiba kihatással van nem csak a fals negatív, de a fals pozitív esetekre, és adott esetben ezekből többre is. A kiértékelés során a szegmentált mondatokat listákként kezeltük, tehát nem vettünk figyelembe semmilyen részszöveg alapú egyezést. Ebben az esetben lehetőség nyílik az F_1 érték meghatározására az alábbi egyenlet alapján:

$$F_1 = 2 \cdot \frac{TP}{TP + FN + TP + FP} = 2 \cdot \frac{TP}{len(y_{true}) + len(y_{pred})} \tag{1}$$

ahol TP a valós pozitív, FP a fals pozitív és FN a fals negatív mondatok számát jelölik, $len(y_{true})$ a valós sztenderd adat elemszámát jelenti, míg $len(y_{pred})$ a prediktált mondatok számát jelenti. Azért tehető meg ez az egyszerűsítés, mivel ebben a problémában nem értelmezhető a valós negatív (TN) elemek száma, és így $len(y_{true}) = TP + FN$ és $len(y_{pred}) = TP + FP$. A TP elemeket mindkét listában pontosan ugyanúgy szereplő mondatok adták. A fentiek miatt a fedés (recall, vagyis hogy az elvárt mondatokból mennyit talál meg a címkéző) és F_1 értékeket számoltuk ki a mintakorpuszokon, melyeket a 4. és az 5. táblázat mutat be. Az adott halmazon elért legjobb eredményt **kiemelés** jelzi. Általánosságban elmondható, hogy a HuSpaCy modelljei jól teljesítettek, és több korpusz esetében ezek bizonyultak a legjobb modellnek. Ez annak fényében nem meglepő, hogy az összes itt felsorolt HuSpaCy modell a teljes Szeged Treebank korpuszon tanult (Szabó és mtsai, 2023; Orosz és mtsai, 2022, 2023). Érdemes továbbá megjegyezni, hogy a `quntoken` sok szempontból más tokenizálási elveket követ, mint a Szeged Treebank (lásd Mittelholtz (2017) 5. fejezet), ami nyilvánvaló visszaesést fog eredményezni, ha azon tesztelik. A `montana-splitter` az összesített

4. táblázat. Fedés értékek a mintakorpuszokon

Adathalmaz	montana-splitter	quntoken	stanza	huspacy_md	huspacy_lg	huspacy_trf	sentence-splitter
ud-test	96,66%	98,66%	97,10%	97,55%	97,77%	**99,11%**	84,19%
fogalmazások	96,59%	98,92%	96,66%	98,42%	98,25%	**99,01%**	97,00%
jogi szövegek	86,69%	86,55%	83,27%	91,82%	86,82%	**95,13%**	84,88%
rövidhírek	94,77%	97,35%	96,49%	96,71%	95,47%	**99,51%**	97,89%
számítástechnikai szövegek	79,08%	85,82%	85,17%	92,63%	88,71%	**94,16%**	80,33%
szépirodalom	85,25%	87,55%	82,86%	89,65%	89,05%	**89,67%**	66,54%
újságcikkek	90,62%	96,48%	94,54%	96,68%	95,17%	**98,17%**	86,85%
Szeged Treebank	88,10%	91,34%	88,73%	93,58%	91,67%	**95,01%**	82,60%
büntető	**95,89%**	85,35%	90,41%	89,93%	88,09%	92,62%	92,09%
gazdasági	**96,25%**	86,63%	90,50%	94,43%	91,73%	94,55%	93,14%
közigazgatási	**98,52%**	82,08%	93,62%	93,24%	82,19%	96,81%	92,52%
munkaügyi	**98,70%**	86,49%	94,29%	94,91%	91,90%	96,94%	94,44%
polgári	**97,20%**	92,16%	92,67%	94,50%	91,35%	94,81%	93,49%
HuCoDe	**97,41%**	86,87%	92,47%	93,51%	89,20%	95,31%	93,25%

Szeged Treebank-en a `sentence-splitter`t előzte meg, és a `stanza` teljesítményét volt képes megközelíteni mindkét metrikában, amely szintén tanult a Szeged Treebank újságcikkjein, ami némiképp befolyásolhatja a korrekt kiértékelést. Érdekesség, hogy a jogi doménen fedés szerint a negyedik, F_1 szerint pedig a harmadik legjobb eredményt érte el a megközelítésünk, 8,44% illetve 5,16%-kal elmaradva a legjobb szegmentálótól. Fontos azonban kiemelni, hogy a HuSpaCy modelleket nem számolva a mi modellünk teljesített a legjobban azon modellek közt, amelyek nem tanultak a jogi alkorpuszon. Általánosságban jobban teljesített a megközelítésünk, mint az eredeti `sentence-splitter` megoldás, annak ellenére, hogy ez utóbbi a rövidhírek esetében a második legjobbnak bizonyult, tehát a fentebb említett jellemző hibák kiküszöbölésére tett erőfeszítéseink nem voltak hiábavalók.

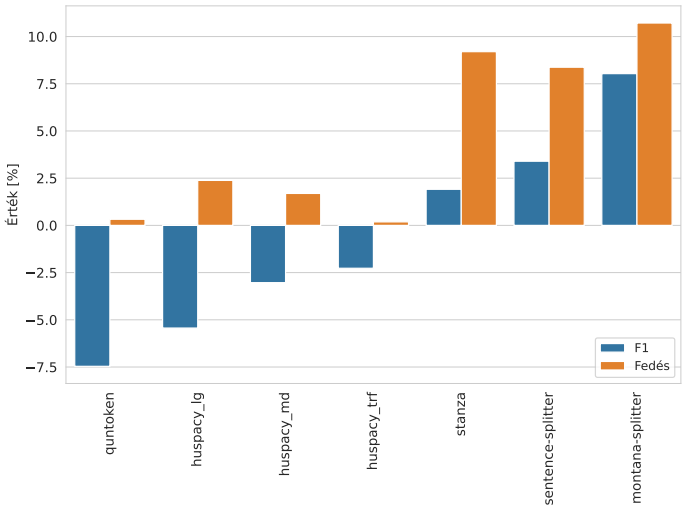
5. táblázat. F_1 értékek a mintakorpuszokon

Adathalmaz	montana-splitter	quntoken	stanza	huspacy_md	huspacy_lg	huspacy_trf	sentence-splitter
ud-test	95,28%	98,55%	97,21%	97,55%	98,21%	**99,00%**	88,52%
fogalmazások	97,29%	**99,12%**	96,79%	98,71%	98,43%	98,89%	97,66%
jogi szövegek	89,42%	87,46%	82,37%	92,62%	88,59%	**94,58%**	85,90%
rövidhírek	95,89%	97,39%	96,96%	97,45%	96,36%	**99,53%**	98,24%
számítástechnikai szövegek	83,67%	88,56%	87,78%	93,83%	90,35%	**95,01%**	84,45%
szépirodalom	85,41%	89,20%	87,03%	90,31%	91,56%	**91,99%**	74,63%
újságcikkek	91,16%	96,37%	95,34%	96,93%	95,53%	**98,08%**	89,54%
Szeged Treebank	89,59%	92,40%	90,43%	94,25%	93,12%	**95,72%**	86,59%
büntető	**95,81%**	77,65%	82,87%	83,02%	80,16%	87,46%	87,07%
gazdasági	**96,50%**	79,35%	72,99%	91,04%	86,70%	91,10%	89,19%
közigazgatási	**98,57%**	73,86%	86,71%	89,71%	74,01%	94,86%	88,28%
munkaügyi	**98,45%**	79,00%	90,28%	92,16%	87,60%	95,20%	90,86%
polgári	**97,15%**	88,64%	88,16%	91,50%	86,82%	91,98%	90,23%
HuCoDe	**97,45%**	80,00%	84,28%	89,59%	83,15%	92,31%	89,30%

A HuCoDe-on összességében és minden jogterület esetében is a `montana-splitter` megoldás bizonyult a legjobbnak, számottevően pontosabb működést felmutatva, mint a többi megoldás. Talán a legmeglepőbb eredmény a `quntoken`hez fűződik: mind a fedés, mind az F_1 metrikában ez érte el a leggyengébb eredményt; a legjobb megoldáshoz képest 10,54% illetve 17,45%-os lemaradással. Érdekes módon a `huspacy_lg` rosszabbul teljesített, mint a `huspacy_md` megközelítés, fedésben 4,31%-kal, F_1 értékben pedig 6,44%-kal. F_1 értékben a második legjobban a `huspacy_trf` megoldás teljesített, de ez is jelentősen, kb. 5%-kal elmaradt a `montana-splitter` teljesítményétől. Megvizsgáltuk, hogy az egyes szegmentálók milyen jellemző hibákat vétettek, a hibaanalízis során a következőket tártuk föl:

– előfordult, hogy a `quntoken` módosította az eredeti szöveget, feltehetőleg karakterenkódolási hiba miatt: -→
– az összes HuSpaCy modell hajlamos volt az idézőjeleket külön mondatként értelmezni,
– a legjellemzőbb hibázási pontok azonban a jogszabály-hivatkozások, az anonimizálással módosított szövegrészek, illetve az egyéb ítélkezési gyakorlatra történő hivatkozások voltak.

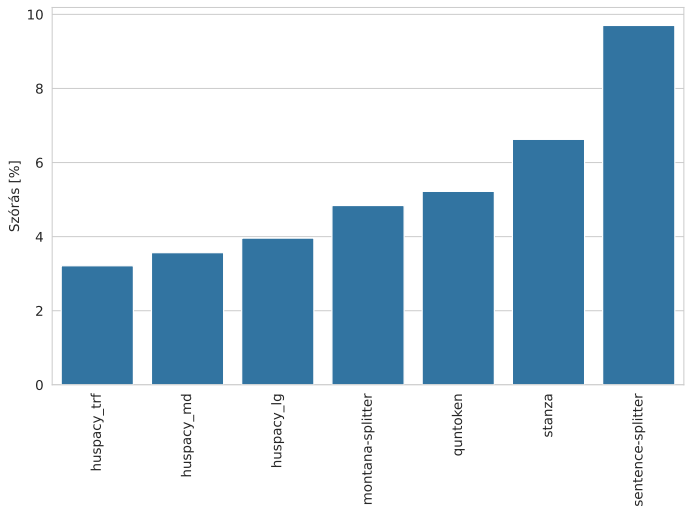
Megvizsgáltuk, hogy hogyan viszonyul egymáshoz az egyes szegmentálók fedés és F_1 eredménye a Szeged Treebank jogi alkorpuszán, valamint a HuCoDe korpuszon, hogy az egyes szegmentálók aldoménfüggéséről kapjunk képet. Az eredményeket az 1. ábra mutatja be.



1. ábra: Az egyes szegmentálási megoldások aldoméntől való függése, a HuCoDe és a Szeged Treebank jogi alkorpusza közti különbség, fedés és F_1 értékben

Látható, hogy a fedés tekintetében a `quntoken` és a `huspacy_trf` is hasonlóan teljesített mindkét korpuszon, minimális, 1% alatti különbségeket tapasztaltunk. Nagyobb, 2% körüli különbséget mértünk a `huspacy_md` és `huspacy_lg` esetekben. Jelentősebb volt a különbség a `sentence-splitter`, a `stanza` és a `montana-splitter` felbontók esetében, ezek jelentősen, 8-10 százalékponttal jobb fedéssel bírtak a bírósági határozatos korpuszon. Az F_1 értékek esetében a `quntoken`, és az összes HuSpaCy szegmentáló legalább 2 ponttal rosszabb értéket ért el a bírósági határozatokon. Ezzel szemben a `sentence-splitter` és a `stanza` szegmentálók 2-3 ponttal jobban teljesítettek, de itt is kiemelkedett a `montana-splitter` megoldás, ami közel 9 ponttal szerepelt jobban. Ezek alapján a legnagyobb jogi doménfüggést a `montana-splitter` tudhatja magáénak, míg a legkiegyensúlyozottabb választásnak a `huspacy_trf` bizonyult.

Megvizsgáltuk a szegmentálók doménfüggését is, kizárólag a Szeged Treebank alkorpuszait felhasználva. Az ezeken kapott F_1 eredmények szórásait mutatja be a 2. ábra mondatszegmentálónként.



2. ábra: Az egyes szegmentálási megoldások doméntől való függése, Szeged Treebank F_1 értékek szórása

Látható, hogy a három legkisebb szórást a három HuSpaCy modell tudhatta magáénak, és leginkább a `sentence-splitter` mutatott doménfüggést, azonban fontos kiemelni, hogy ezen a korpuszon mindhárom HuSpaCy modell tanult, így az sem meglepő, hogy ezek teljesítettek a legkiegyensúlyozottabban. Ezzel szemben a `montana-splitter` szegmentáló a doménfüggést tekintve a középmezőnyhöz tartozott. Az, hogy a `montana-splitter` esetén a szórás jelentősen csökkent a `sentence-splitter`hez képest, jól prezentálja a szabályalapú megoldás testre szabhatóságát.

A fentebb bemutatott eredmények rámutatnak arra, hogy a mondatszegmentálás minősége jelentősen eltérő lehet a különböző domének esetében, így körültekintőnek kell lenni, amikor a mondatok felbontását végezzük, mert az a szabályszerűség, amely egy korpuszra jellemző, nem biztos, hogy a másikra is igaz. Erre a legjobb példa a jelen cikkben bemutatott szegmentálónk teljesítménye a Szeged Treebank jogi alkorpuszán. Habár látszólag a két domén rendkívül hasonló, a bírósági határozatok mégis alapvetően más jellemzőkkel bírnak, mint a jogszabályok szövegei.

5.2. A tördelés ismeretének hatása

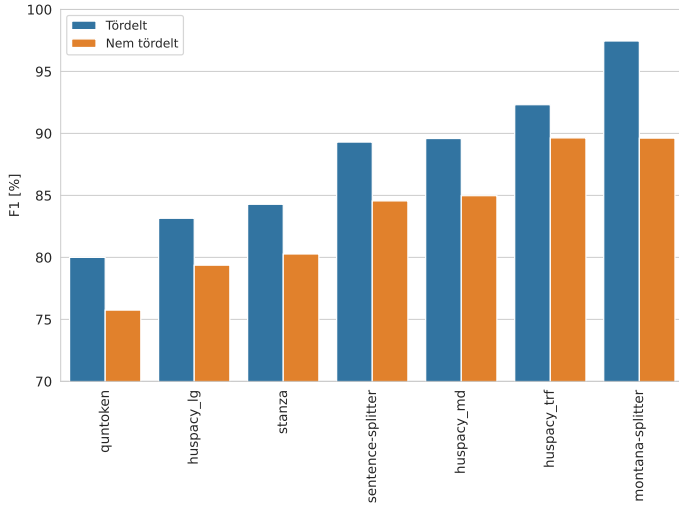
Mivel az általunk készített korpusz tartalmazza a tördeléssel kapcsolatos információkat is, megvizsgáltuk, hogy ennek milyen hatása van az egyes szegmentálók megbízhatóságára. Először megvizsgáltuk, hogy mi lenne, ha a mondatokat csupán a határozatokban levő új sorok segítségével akarnánk meghatározni (egy sor egy mondat) és így milyen fedést és F_1 értéket lehetne ezzel a módszerrel elérni. A kiértékelés előtt kiszűrtük a felbontás után keletkező üres mondatokat. Az eredményeket a 6. táblázat tartalmazza.

6. táblázat. Bekezdés alapú mondatszegmentálás a bírósági határozatok korpuszon

Jogterület	**Büntető**	**Gazdasági**	**Közigazgatási**	**Munkaügyi**	**Polgári**
Fedés [%]	32,84	31,79	29,98	27,22	32,34
F_1 [%]	40,21	41,42	39,06	35,89	41,46

Látható, hogy minden jogterület esetében viszonylag nagy volt az olyan mondatok száma, amelyeket új sor is követett. Ha ezt a megközelítést mondatszegmentálóként használnánk, 30% körüli fedést tudna, 40% körüli F_1 értékkel a HuCoDe korpuszon.

Ezt követően megvizsgáltuk, hogy hogyan teljesítenek az egyes szegmentálók akkor, ha az új sorokra vonatkozó információk már nem állnak rendelkezésükre. Az eredményeket a 3. ábra mutatja be. Látható, hogy nem volt olyan módszer, amire ne hatott volna pozitívan az, hogy ismeri a tördelést a szövegben. Az is látható azonban, hogy ezen információból elsősorban a `montana-splitter` megközelítés profitált igazán. Ezzel szemben a `huspacy_trf` módszerre volt a legkevesebb hatással ezen információ megléte. Ez az eredmény nem meglepő a transzformer alapú megközelítések lényegesen szofisztikáltabb nyelvismeretét figyelembe véve. Fontos kiemelni, hogy a tördelés nélküli adaton a `montana-splitter` és a `huspacy_trf` is közel ugyanúgy teljesített F_1 szerint. Tehát sikerült egy olyan szabályalapú megoldást kifejlesztenünk, amely a tördelést tartalmazó bírósági határozatokon lényegesen jobban teljesített, mint a legkomolyabb architektúrával bíró transzformer alapú mondatszegmentáló, és a tördelés nélküli esetben is ugyanolyan pontosan volt képes működni.



3. ábra: Mondatszegmentálók F_1 értékének tördeléstől való függése

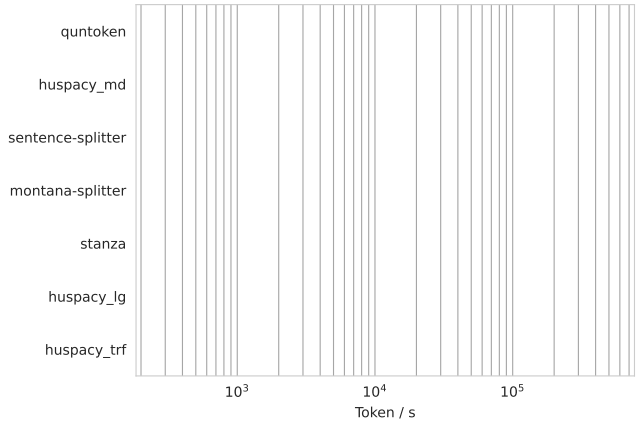
5.3. Futásidő

A gyakorlatban a pontos működés mellett a másik fontos szempont, hogy milyen futásidővel lehet számolni az egyes megoldások esetében. A 4. ábra az 1 másodperc alatt átlagosan feldolgozott tokenek számát mutatja be, tokenként a whitespace-ek (szóközök, új sorok, és ezek speciális verziói) mentén felbontott és a nem üres tokeneket értjük. A méréseket ugyanazon a gépen végeztük el, amely 8 CPU maggal, és 20 GB memóriával rendelkezett, GPU-val azonban nem.

Látható, hogy messze a leggyorsabban a `quntoken` volt képes feldolgozni az adatokat, jelentősen gyorsabbnak bizonyult ez a megoldás, mint bármelyik másik: közel két nagyságrendi különbség volt tapasztalható a második leggyorsabbhoz képest is. A második leggyorsabbnak a `huspacy_md` bizonyult, amely jól teljesített a fedés és F_1 metrikákban is. A megoldásunk a negyedik leggyorsabb volt, közvetlenül a `sentence-splitter` megoldást követően, jelentősebb előnnyel megelőzve a `stanza` és `huspacy_lg` és `huspacy_trf` megoldásokat. Így, figyelembe véve a futásidőt és a pontos működést, a bírósági határozatok mondatszegmentálására összességében kijelenthető, hogy a szegmentálónk a legjobb választás.

6. Összefoglalás

A cikkben bemutattunk egy bírósági határozatok szegmentálására „hangolt“ szabályalapú mondatszegmentálót, amelyet összemértünk a jelenleg magyar nyelvben széleskörűen használt mondatszegmentálók teljesítményével. A kiértékeléshez a Szeged Treebank 2.0 és UD korpuszokat használtuk fel, valamint a bírósági határozatokon való összehasonlításhoz magyar bírósági határozatokból



4. ábra: A szegmentálók teljesítménye a mintakorpuszokon. A tokenszámot whitespace felbontással határoztuk meg.

képeztünk korpuszt, melynek lépéseit szintén bemutattuk. A megközelítésünk a Szeged Treebank korpuszon összességében a `stanza` szegmentálóval hasonló eredményt ért el, attól némileg elmaradva, azonban a jogi alkorpuszon csak a HuSpaCy szegmentálók voltak képesek jobban teljesíteni, amelyek tanultak ezen a részkorpuszon. A bírósági határozatokat tartalmazó korpuszon a tördelés ismeretében a szabályalapú mondatszegmentálónk teljesített legjobban, jelentős, 5% fölötti F_1-beli különbséggel megelőzve a második legjobb, `huspacy_trf` megoldást. A tördelések törlésével ez a különbség csökkent, de még így is képes volt a megoldásunk pontosan ugyanúgy teljesíteni F_1 értékben, mint a `huspacy_trf` szegmentáló. A tördelés meglétének szegmentálókra való hatását vizsgálva a HuCoDe korpuszon kijelenthető, hogy annak ismeretében minden szegmentáló jobban volt képes teljesíteni. További tapasztalat volt, hogy a bírósági határozatokon leginkább a mi szegmentálónk profitált ennek ismeretéből, legkevésbé pedig a HuSpaCy transzformer alapú szegmentálója. Az eredményekből megállapítható volt továbbá, hogy a domén, de az aldomén is jelentősen befolyásolhatja a szegmentálók teljesítményét, a szabályalapú megoldás hangolásával azonban ez a függés jelentősen csökkenthetőnek bizonyult.

Összefoglalva tehát sikerült egy olyan szabályalapú megoldást kifejlesztenünk, amely a tördelést tartalmazó bírósági határozatokon lényegesen jobban teljesített, mint a legkomolyabb architektúrával bíró transzformer alapú mondatszegmentáló, és a tördelés nélküli esetben pedig ugyanolyan pontosan volt képes működni.

Köszönetnyilvánítás

Szeretnénk köszönetünket nyivánítani a Wolters Kluwer Hungary Kft. munkatársainak a kézi mondatszegmentálásban elvégzett munkájukért.

Hivatkozások

Csányi, G., Gadó, K., Bajári, L., Megyeri, A., Fülöp, A., Egri, E., Vági, R., Nagy, D., Vadász, J.P., Üveges, I.: Mondatszám-meghatározás hatása a magyar nyelvű jogi szövegek extraktív kivonatainak minőségére. In: IX. Magyar Számítógépes Nyelvészeti Konferencia. pp. 77–90 (2023)

Csendes, D., Csirik, J., Gyimóthy, T., Kocsor, A.: The Szeged Treebank. In: Text, Speech and Dialogue: 8th International Conference, TSD 2005, Karlovy Vary, Czech Republic, September 12-15, 2005. Proceedings 8. pp. 123–131. Springer (2005)

De Marneffe, M.C., Manning, C.D., Nivre, J., Zeman, D.: Universal dependencies. Computational linguistics 47(2), 255–308 (2021)

Ghosh, S., Wyner, A.: Identification of Rhetorical Roles of Sentences in Indian Legal Judgments. Legal Knowledge and Information Systems: JURIX p. 3 (2019)

Laki, L.J., Yang, Z.G.: Neural machine translation for Hungarian. Acta Linguistica Academica 69(4), 501–520 (2022)

Mittelholtz, I.: emToken: Unicode-képes tokenizáló magyar nyelvre. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia. pp. 61–69 (2017)

Orosz, G., Szabó, G., Berkecz, P., Szántó, Z., Farkas, R.: Advancing Hungarian Text Processing with HuSpaCy: Efficient and Accurate NLP Pipelines. In: Text, Speech, and Dialogue. pp. 58–69. Springer Nature Switzerland (2023)

Orosz, G., Szántó, Z., Berkecz, P., Szabó, G., Farkas, R.: HuSpaCy: an industrial-strength Hungarian natural language processing toolkit. In: XVIII. Magyar Számítógépes Nyelvészeti Konferencia. pp. 59–73 (2022)

Parikh, V., Mathur, V., Mehta, P., Mittal, N., Majumder, P.: Lawsum: A weakly supervised approach for indian legal document summarization. arXiv preprint arXiv:2110.01188 (2021)

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., Manning, C.D.: Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations (2020)

Santosh, T.S., Bock, P., Grabmair, M.: Joint Span Segmentation and Rhetorical Role Labeling with Data Augmentation for Legal Documents. In: European Conference on Information Retrieval. pp. 627–636. Springer (2023)

Szabó, G., Orosz, G., Szántó, Z., Berkecz, P., Farkas, R.: Transformer-alapú HuSpaCy előelemző láncok. In: XIX. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2023). pp. 305–317 (2023)

Üveges, I.: Comprehensibility and Automation: Plain Language in the Era of Digitalization. TalTech Journal of European Studies 12(2), 64–86 (2022), https://doi.org/10.2478/bjes-2022-0012

Váradi, T., Simon, E., Sass, B., Mittelholcz, I., Novák, A., Indig, B.: E-magyar–A Digital Language Processing System. European Language Resources Association (ELRA) (2018)

Yang, Z.G., Agócs, Á., Kusper, G., Váradi, T.: Abstractive text summarization for Hungarian. In: Annales Mathematicae et Informaticae. vol. 53, pp. 299–316. Eszterházy Károly Egyetem Líceum Kiadó (2021)