Automatikus tematikuscímke-ajánló rendszer sajtószövegekhez

Yang Zijian Győző[1,2], Novák Attila[1,2], Laki László János[1,2]

[1]Pázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar
[2]MTA-PPKE Magyar Nyelvtechnológiai Kutatócsoport
1083 Budapest, Práter u. 50/a.
{yang.zijian.gyozo, novak.attila, laki.laszlo}@itk.ppke.hu

Kivonat Cikkünkben sajtószövegek automatikus tematikus címkézésével kapcsolatos kutatásunk eredményét, illetve a kutatás keretében létrehozott automatikus címkézőrendszert mutatjuk be. A rendszerhez olyan felhasználói felületet hoztunk létre, amely lehetővé teszi a felhasználó számára a rendszer bizonyos paramétereinek beállítását. Ennek segítségével az ajánlott kulcsszólista fedése és pontossága testre szabható. Bemutatjuk a különböző modellparaméterek beállításának hatását a címkézés teljesítményére.
Kulcsszavak: címkeajánlás, kulcsszavazás, fastText, SentencePiece tokenizáló

1. Bevezetés

A webes hírportálokon megjelenő szövegeket gyakran különböző tematikus címkékkel látják el. Ezek szerepe többféle. Egyrészt lehetővé teszik a látogatók számára, hogy kifejezetten egy-egy számukra érdekes témával, személlyel, eszközzel stb. kapcsolatos cikkeket vagy egyéb tartalmakat a kulcsszavakra (tematikus címkékre) szűrve megjelenítsék. Másrészt a kulcsszavakat az adott cikkhez kapcsolódó egyéb cikkek vagy tartalmak szűréséhez/megjelenítéséhez is használják. Emellett a tematikus kulcsszavak arra is használhatóak, hogy az üzemeltető regisztrált vagy egyéb módon nyilvántartott felhasználói számára az érdeklődésüknek megfelelő testre szabott tartalmakat ajánljon.

Szerepet játszanak a címkék a keresőmotorok (pl. a Google) találatrangsorolási algoritmusaiban is. A html tartalom megfelelő kulcsszó-metacímkéinek tartalmaként megadott kulcsszavakat a keresőmotorok korábban egyértelműen előrébb rangsorolták, mint a puszta szövegszótalálatokatA forgalomnövelés céljából végrehajtott manipulatív keresőoptimalizálás céljából bevezetett sok hamis címke megjelenése miatt a keresőmotorok üzemeltetői később csökkentették vagy mellőzni kezdték a kulcsszó-metacímkék tartalmának figyelembevételét a találatok rangsorolásánál.

Ennek ellenére a korábban felsorolt okokból, illetve mert a valóban releváns kulcsszavak továbbra is fontosak, illetve a kulcsszavak köré rendezett tematikus cikkgyűjteményoldalakat a keresőmotorok továbbra is lelkesen indexelik (itt a

cím, illetve az url része az adott kulcsszó), a megfelelő kulcsszavak cikkekhez rendelése továbbra is fontos az online sajtó számára. Ugyanakkor bár az adott szöveghez kapcsolódó tematikus kulcsszavak automatikus hozzárendelésére számos algoritmikus megoldás létezik, sok online is megjelenő szövegarchívumban a kulcsszavak tartalomhoz való hozzárendelését jelenleg is kizárólag emberi munkával végzik.

Van, ahol központilag egyetlen – tipikusan könyvtáros végzettségű – munkatárs végzi a cikkek kulcsszavazását. Ez a megoldás viszonylag egységes és jól átgondolt címkehasználatot eredményez, azonban a hosszú átfutási idő miatt – a címkézőnek el kell olvasnia minden cikket – csak viszonylag korlátozott mennyiségű tartalom címkézése oldható meg így. Ez a módszer egy hetilap esetében alkalmazható, azonban egy adott idő alatt jóval nagyobb mennyiségű – ráadásul online karbantartott, illetve időnként módosított – tartalmat generáló webes tartalomszolgáltató, illetve hírportál esetében nincs idő arra, hogy egyetlen dedikált személy végezze a címkézést. Ebben az esetben a manuális címkézést maguk a szerzők végzik, és az egységesség irányába csak a szerkesztőségi irányelvek, illetve esetlegesen a szerkesztőség által használt tartalomkezelő rendszerbe (CMS) integrált prediktív keresésen alapuló ajánló használata mutat (ahogy a szerző elkezdi gépelni a kulcsszót, a korábban már használt egyező kezdetű kulcsszavak listája megjelenik, és a lista tovább gépelve egyre szűkül). Ugyanakkor egyrészt minden szerző bármikor új kulcsszót vehet fel, másrészt a prediktív ajánló használata egy félregépelt kulcsszó gyakori használatához is vezethet.

Cikkünkben egy olyan tartalomkezelő rendszerekbe integrálható rendszert mutatunk be, amely automatikus kulcsszóajánlással segíti a szerkesztőség munkáját. A korábban manuálisan címkézett szövegeken betanított modellt használunk az újonnan születő szövegekhez alkalmazható címkék megjósolására.

2. Kapcsolódó irodalom

Magyar szövegek tematikus kulcsszavazásával kapcsolatban fontos korábbi eredményről számol be Farkas (2009). Ebben a cikkben az [origo] hírportál korábbi kulcsszavakkal el nem látott tartalmainak teljesen automatikus címkézésére alkalmazott megoldást mutat be a szerző. Megoldásuk igen sokrétű: a cikkek szövegét különböző szintű elemzésnek (szófaji és névelemcímkézés) vetették alá, névszói csoportokat kerestek, azokat az egységes címkekészlet létrehozása érdekében normalizálták/lemmatizálták, emellett a címkéket különböző osztályokba (személy, földrajzi név, szervezet, egyéb téma) sorolták. A beszámoló ugyanakkor nem említi, hogy a rendszert az újonnan írt cikkek címkézésének segítésére használták volna. Az akkori fejlesztés során viszonylag kevés címkézett szöveg állt rendelkezésre, ezért elsősorban az adott szövegben szereplő kifejezések kiemelésére és normalizálására támaszkodtak a címkézéskor.

Mi abból indultunk ki, hogy nagyméretű manuálisan címkézett szöveghamaz áll rendelkezésre, ezért elsősorban ennek kiaknázására alapoztuk megoldásunkat, amelynek elsődleges célja a további alapvetően továbbra is emberi kontroll alatt végzett címkézési munka hatékony segítése. Választásunk a fastText programcso-

mag (Joulin és mtsai, 2017) címkézőalgoritmusára esett. Ebben a megoldásban a neurális osztályozóhálózat(ok) bemenetén az adott szöveg tokenjeinek, illetve token-n-eseinek reprezentációja jelenik meg (a bennük szereplő különböző hosszú karakter-n-gramok reprezentációjának eredőjeként), és az osztályozó ehhez a szövegreprezentációhoz és az egyes lehetséges címkékhez rendel illeszkedési értéket multinomiális logisztikus regresszió alkalmazásával. Megfelelő küszöbérték kiválasztásával az adott szövegre jól illeszkedő kulcsszavak elválaszthatóak a kevésbé jól illeszkedőektől. Bár megjelenése óta a fastText modellnél jobban teljesítő szövegosztályozó modellek is megjelentek (a cikk írásának idején az ilyen jellegű feladatokban az XLNet architektúra adja a legjobb teljesítményt nyújtó megoldást több angol nyelvű adatbázison (Yang és mtsai, 2019)), ezeknek komplexitása, hardver- és futásiidő-igénye a pontosságbeli teljesítménykülönbséget jóval meghaladó mértékben nagyobb, mint a fastTexté.

A fastText által előállított szóbeágyazási modell osztályozási feladatra való alkalmazását mutatja be Szántó és mtsai (2017), azonban az ott bemutatott osztályozási feladat egyszerű kétosztályos osztályozást jelentett (sport/videojáték) szemben az általunk kitűzött céllal, ahol sok ezer, sőt akár sok tízezer lehetséges egyedi címke közül kell kiválasztani az adott szövegre legjobban illeszkedő címkéket (amelyek száma a témától és a szöveg hosszától függően változhat).

3. A címkézőrendszer

Ebben a részben áttekintjük a címkézőrendszer architektúráját, illetve bemutatjuk röviden a mögöttes adatbázist.

3.1. A címkézőrendszer architektúrája

Az elkészült címkézőrendszer[1] egy REST alapú webes applikáció, ami két részből áll: frontend és backend. A frontend egy javascript bootstrap alapú felület, amin a felhasználó egy formon keresztül be tudja vinni az adatokat (cikk szövege, lead, cím, szerző, évszám), majd a *Címkézés* gombbal le tudja kérni a címkéző által javasolt címkéket, melyeket a felület megjelenít. A felhasználónak lehetősége van kiválasztani, hogy melyik modellel szeretné címkézni a cikket.

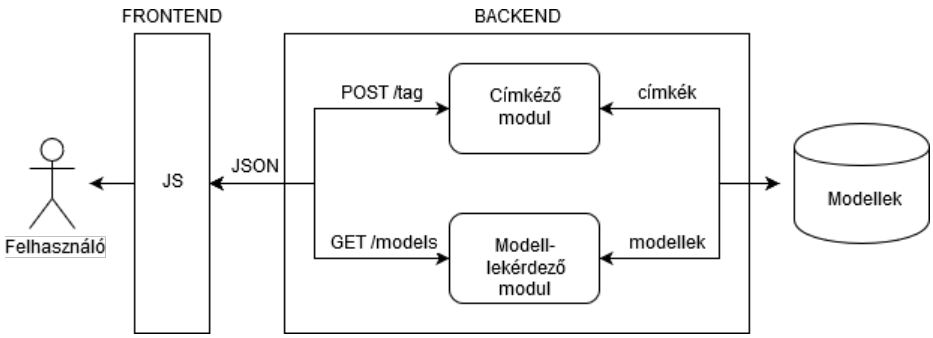
A megjelenített címkéknek három kategóriája van: kulcsszavak, tulajdonnevek és trendcímkék. A trendcímkék olyan címkék, amelyek valamilyen konkrét egyedi eseményre (egy konkrét választás, fesztivál, olimpia, konferencia stb.) utalnak, és inkább rövid távon van jelentőségük és értelmük. Ilyen például az *Oscar-gála 2018,* amely ugyan meglehetősen hasonló bármely más adott évben lezajlott Oscar-gálához, azonban az adott díjátadó időpontját övező viszonylag behatárolt időszakon kívül feltehetőleg nem hasznos címke. A kulcsszavak és tulajdonnevek statikus címkék, nem időszakhoz vagy konkrét eseményhez kötöttek. Ilyen például a *politika*, a *zene* stb., illetve a személyt, szervezetet stb. jelölő névcímkék mellett az általánosabb eseménytípust jelölő név jellegű címkék, mint pl. a dátum nélküli *Oscar-gála*.

[1] http://nlpg.itk.ppke.hu/projects/tagger

A felületen továbbá lehetőség van egy 0-tól 1-ig terjedő skálán beállítani, hogy milyen valószínűségű/konfidenciaértékű címkéket jelenítsen meg, valamint van egy „minimum 3” funkció, amelynek bekapcsolása esetén, függetlenül a konfidenciaküszöbtől, mindenképpen minimum 3 darab címkét jelenít meg.

A tesztanyagunknak készítettünk egy külön demófelületet, amelyre a címkézőfelületről át lehet navigálni. A tesztanyaghoz tartozó felület megegyezik a demófelülettel, annyi kiegészítéssel, hogy láthatóak a tesztkorpuszban szereplő szövegekhez tartozó referenciacímkék is, valamint megjeleníti a pontosság és a fedés értékeit is.

A címkéző másik része a backend. A frontend AJAX kéréssel tudja beküldeni a cikket és lekérni a backendtől az ajánlott címkéket. A frontend és a backend egymás között JSON formátumú adatcsomagokkal kommunikálnak. A backend egy Python nyelven implementált Flask[2] webszerver. A webszerver indításkor betölti az előre betanított címkézőmodelleket. Külön modellt hoztunk létre a statikus, a trend- és a tulajdonnévcímkékre. Emellett az online adatbázis cikkeiből tanított és a nyomtatott és az online adatbázis összevont anyagán betanított modell is kipróbálható. A modelleket rendszeresen újratanítjuk, hogy a szerzők által újonnan felvitt címkék bekerüljenek a rendszerbe. A trendcímkemodell tanításakor csak az utolsó fél év anyagát használjuk. A régebbi szövegeknél a trendcímkéket azok általános ekvivalensére cseréljük le.



1. ábra: A címkézőrendszer architektúrája

3.2. A korpusz

Tanító és tesztkorpuszként a hvg.hu által rendelkezésünkre bocsátott nyomtatott és online hírlapból vett cikkeket használtunk fel. Első kísérleteinket az 1. táblázatban bemutatott korpuszon végeztük. Később bővebb anyaghoz jutottunk: az online cikkadatbázis kiegészült a 2012 és 2016 közötti anyaggal. Az utóbbi bővített korpuszon végzett kísérleteinket a 6. részben mutatjuk be.

[2] https://palletsprojects.com/p/flask

Kísérleteink során többféle tanítóanyagot hoztunk létre: a nyomtatott hetilap anyagából, az online cikkekből, illetve a kettőt ötvöző hibrid korpuszt. A három különböző tanítóanyagból további változatokat hoztunk létre. Kísérleteztünk kisbetűsítéssel, stopword-listában szereplő szavak törlésével, ezek kombinációjával, valamint a szövegeken betanított SentencePiece tokenizáló alkalmazásával.

	Nyomtatott hírlap	Online hírlap
Felépítés	id, cím, lead, cikk szövege, szerző kategória, év, dátum	id, cím, alcím, cikk szövege, szerző, kategória, létrehozás és módosítás dátuma
Címkefajták	kulcsszavak, személy, szervezet, földrajzi	kulcsszavak
Megjelenés	hetilap	napilap
Időszak	1994-2017	2017-2018
Témák	gazdaság, politika, tudomány, sport kultúra, pszichológia	gazdaság, politika, tudomány, sport kultúra, pszichológia, blog,
Statisztika	cikk: 119077 cikk címke: 1023 db token: ∼73 millió type: 62 ezer	cikk: 86256 cikk címke: 76654 db token: ∼35 millió type: ∼33 ezer

1. táblázat. Nyomtatott és online hírlap tulajdonságai

3.3. Használt eszközök

Modelljeinket a fastText programkönyvtár a SentencePiece tokenizáló használatával készítettük.

A fastText (Joulin és mtsai, 2017) egy nyílt forráskódú programkönyvtár osztályozási feladatra és szövegreprezentációs modell létrehozására. Az eszközt a Facebook fejlesztette C++ nyelven.

A SentencePiece nevű eszköz egy felügyelet nélküli szövegtokenizáló és detokenizáló. Implementálva van benne a BPE algoritmus, ami egy unigram nyelvmodellel (Kudo, 2018) van súlyozva. Használatával elhagyhatók a nyelvspecifikus előfeldolgozási lépések, mint például a tokenizálás vagy a kisbetűsítés. A módszer lényege, hogy a természetes szöveget úgy alakítja át, hogy abban a különböző tokenek száma egy paraméterként megadható korlátos szám legyen, ezért az így létrejött tanítóanyagban általában nem lesznek ismeretlen szavak. (A tanítóanyagban nem szereplő ismeretlen karakterek (pl. idegennyelvű szövegrészekből), s így ismeretlen tokenek ritkán előfordulhatnak.) Ennek köszönhetően a neurális modellek paraméterszáma nagymértékben csökkenthető a hagyományos szóalapú modellekhez képest. A módszer a neurális gépi fordítás területéről származik, és a szövegfeldolgozásra használt mélytanuló modellekben nagyon elterjedt a használata.

4. Kísérletek

Első lépésként az 1. táblázatban látható nyomtatott cikkadatbázissal (NYC) kísérleteztünk. Az alábbi modelleket hoztuk létre:

- NYC-T: tokenizált (T) szöveg
- NYC-TK: tokenizált és kisbetűsített (K) szöveg
- NYC-PKS: Sentence Piece tokenizált (P) és kisbetűsített szöveg; stopwords lista használta
- NYC-PS: Sentcence Piece tokenizált szöveg; stopwords lista használata
- NYC-P: Sentcence Piece tokenizált szöveg

Stopwords listához az NLTK[3] magyar nyelvű csomagjához tartozó stopwordlistát használtuk, amely 199 szót tartalmaz.

Következő lépésként az online cikkadatbázissal (OLC) kísérleteztünk. Az alábbi modelleket hoztuk létre:

- OLC-P: Sentence Piece tokenizált szöveg

5. Eredmények

A 2. táblázatban láthatóak az első lépés eredményei, amelyeket 0,8-as valószínűség mellett értünk el a nyomtatott cikkadatbázis anyagán. Az egyik szembetűnő eredmény, hogy a SentencePiece tokenizáló használata majdnem kétszeresére növelte a fedés értéket. A másik érdekes eredmény, hogy a legnagyobb pontosságot az a modell érte el, amelyik kizárólag SentencePiece tokenizálót használt, sem kisbetűsítést, sem stopwordlistát nem. De az is látható, hogy ha elhagyjuk sorban ezeket az eszközöket, ugyan a pontosság nő, a fedés értéke csökkenni kezd.

Modell	Pontosság	Fedés
NYC-T	0,749	0,152
NYC-TK	0,750	0,162
NYC-PKS	0,748	**0,362**
NYC-PS	0,768	0,300
NYC-P	**0,774**	0,284

2. táblázat. A nyomtatott cikkadatbázissal végzett kísérleteink

A 3. táblázatban látható néhány példa arra, hogy milyen címkéket ajánl a rendszerünk. Látható, hogy amikor a rendszer magas valószínűséggel becsül, azok a címkék majdnem azonosak az eredeti címkékkel. De az is látható, hogy az alacsonyabb pontossággal becsült címkék szintén elég közel állnak a témához, és általában jó ajánlások. Vannak esetek, mint például az „alkotmánybíróság”

[3] https://pythonspot.com/nltk-stop-words/

esetében, hogy csak egy címkét társítottak a cikkhez, ajánlórendszerünk pedig az összes hasonló jelentésű címkét visszaadta eredményül.

Van olyan eset is, amit a „belpolitika" példáján látható, hogy az eredeti címkék között nem szerepel az „önkormányzat", de a cikk tartalmát tekintve erősen összefügg ezzel, ezért az ajánlórendszerünk ezt gondolta legvalószínűbbnek.

A „fúzió" példában, bár a címkék között nem szerepel a sztrájk, de a cikk végén több mondat is szól a sztrájkról, ezért a rendszerünk ajánlja ezt a címkét.

Láthatunk továbbá példát arra, hogy az NYC modell az online cikkeire tesz ajánlást. Teljesítményt nem tudtunk mérni ezen, hiszen teljesen más a címkekészlet. De a példákból láthatjuk, hogy teljesen jól megközelíti a témát. A „stewardess" címke nem szerepel a NYC modellben, de helyette „foglalkoztatás" címkével egész jól közelíti.

6. Részletes kísérletek a címkegyakoriság és a teljesítménymutatók összefüggésével kapcsolatban

Az előző részekben említett előzetes kísérletek után részletes kísérleteket végeztünk a hetilapkorpuszon és a 2012–2018 közötti időszakból származó kibővített online anyagon. Szerettük volna megtudni, hogy milyen összefüggés van az egy-egy címkére a tanítóanyagban látott példák száma és a rendszer teljesítménye között. Mindkét korpuszt úgy osztottuk tanító- és tesztanyagra, hogy a korpuszban legalább 15-ször szereplő címkékre a tesztanyagban legyen legalább 5 példa.

A két korpuszból készített tanító és tesztanyagunk jellemzőit a 4. táblázatban foglaltuk össze. A hetilapkorpusz cikkei hosszabbak, így az online korpusz több mint háromszor annyi cikke szószámban kevesebb mint kétszer akkora terjedelmű. Bár a hetilapkorpusz jóval hosszabb időszakot ölel fel, a címkehasználat egységesebb: a másik korpusz 7,5-szer többféle címkét tartalmaz. Ami a névcímkék arányát illeti (a nagybetűt tartalmazó címkéket soroltuk ide): ezek teszik ki a címkeelőfordulások nagyjából 95%-át. Ugyanakkor a címketípusok (a különböző címkék) jóval nagyobb része név a hetilapkorpuszban (96% a tanítóanyagban), mint az online korpuszban (50% alatt). Ennek oka egyrészt a nem név jellegű tematikus címkék sokkal nagyobb változatossága, másrészt a ritkább névcímkék sokszor kicsit hanyag kisbetűs írásmódja. A név-fogalom homográf párok mindkét tagja (pl. *Bugyi–bugyi, Magyar Csapat–magyar csapat*) az esetek nagy részében összevonva kisbetűvel szerepel címketöbbértelműséget eredményezve. A tesztanyagban igen, de a tantóanyagban nem szereplő (OOV) címkelőfordulások aránya a hetilapkorpusz esetén 2,7%, az online korpusznál 6,9%.

A hetilapkorpusz anyagán tokenizálás nélkül, hagyományos tokenizálással, és a SentencePiece tokenizálóval is betanítottunk egy-egy modellt, az online anyagon csak az utóbbit teszteltük. A tanítás paraméterei azonosak voltak. Minden esetben one-to-many osztályozókat tanítottunk be a szövegenként változó címkeszám kezelésére, és 100 dimenziós vektorokat használtunk. Az online korpusz sokkal nagyobb elemszámú címkekészlete miatt az online modell betanítása jóval hosszabb időt igényelt annak ellenére, hogy a hetilapkorpusz 50 epochos tanításával szemben 30 epoch (teljes korpuszbejárás) volt a tanítás ideje.

Eredeti címkék	Ajánlott címkék
NYC modell: ajánlások a hetilapkorpuszból származó cikkekre	
fúzió, gazdaság, közlekedés légi közlekedés	légi közlekedés (0,999), fúzió (0,743) gazdaság (0,719), közlekedés (0,378) sztrájk (0,033), repülőterek (0,009) vállalatgazdaság (0,003)
alkotmánybíróság	jog (1,000), alkotmányjog (1,000) alkotmánybíróság (1,000), alkotmány (0,772) büntetőjog (0,699), jogtörténet (0,328) jogalkotás (0,197)
belpolitika, pártpolitika, tömegközlekedés	önkormányzat (0,994), belpolitika (0,971), pártpolitika (0,492), tömegközlekedés (0,133), politika (0,104), önkormányzati költségvetés (0,078), uniós támogatás (0,032)
NYC modell: ajánlások az online korpuszból származó cikkekre	
földrengés, irán földmozgás, utórengés	földrengés (1,000), természeti katasztrófa (0,999), katasztrófa (0,998), idősek (0,053) halálbüntetés (0,022), katasztrófavédelem (0,017)
repülés, stewardess, wizz air	légi közlekedés (1,000), repülőgépgyártás (0,160), foglalkoztatás (0,078), járműipar (0,038) repülőterek (0,010), gazdaság (0,002) közlekedés (0,002)
OLC modell: ajánlások az online korpuszból származó cikkekre	
földrengés, irán földmozgás, utórengés	földrengés (1,000), irán (0,979) örményország (0,508), utórengés (0,294) lövöldözés (0,192), halálos áldozat (0,178)
repülés, stewardess, wizz air	wizz air (1,000), légitársaság (0,818) repülés (0,803), lufthansa (0,314) repülőgép (0,307)

3. táblázat. Példák a címkézőrendszer ajánlásaira

korpusz		token				típus	
	cikk	szó	címke	név	OOV	címke	név
Hetilap tanító	94094	46,25M	0,46M	0,44M	-	24849	23822
Hetilap teszt	6902	3,33M	38775	37559	1036	5089	4152
Online tanító	328635	89,08M	1,33M	1,26M	-	186508	89711
Online teszt	45105	13,43M	0,21M	0,2M	14488	53568	24607

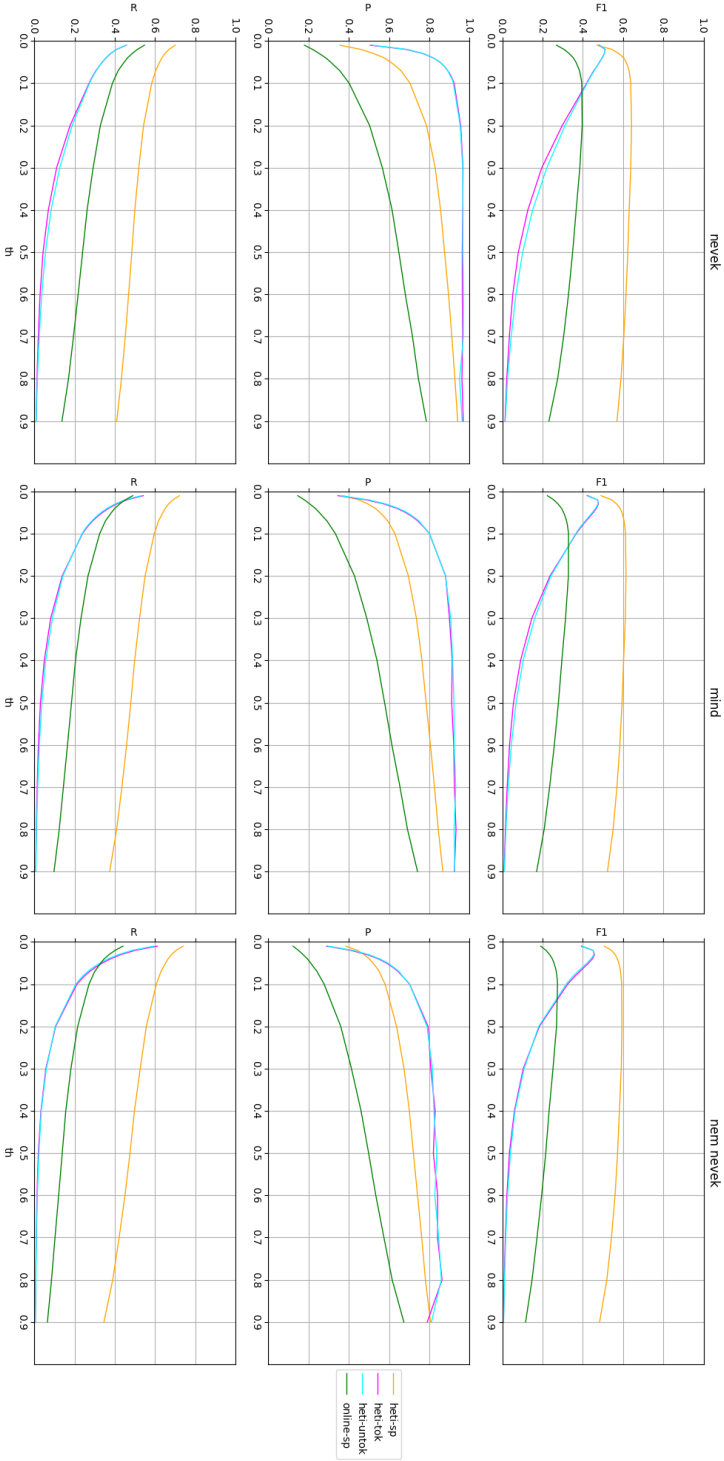
4. táblázat. A modellek betanítására és mérésére használt korpuszok jellemzői

Méréseink során a címkéket csoportokba osztottuk a tanítókorpuszbeli gyakoriságuk szerint. Mértük az egyes gyakorisági osztályokba tartozó címkék pontosságát, fedését és F_1-mértékét a javasolt címkelista különböző konfidenciaszintek melletti vágása esetén. Külön mértük a név-, illetve fogalmi címkékre, valamint az összes címkére vonatkozó eredő teljesítményt.

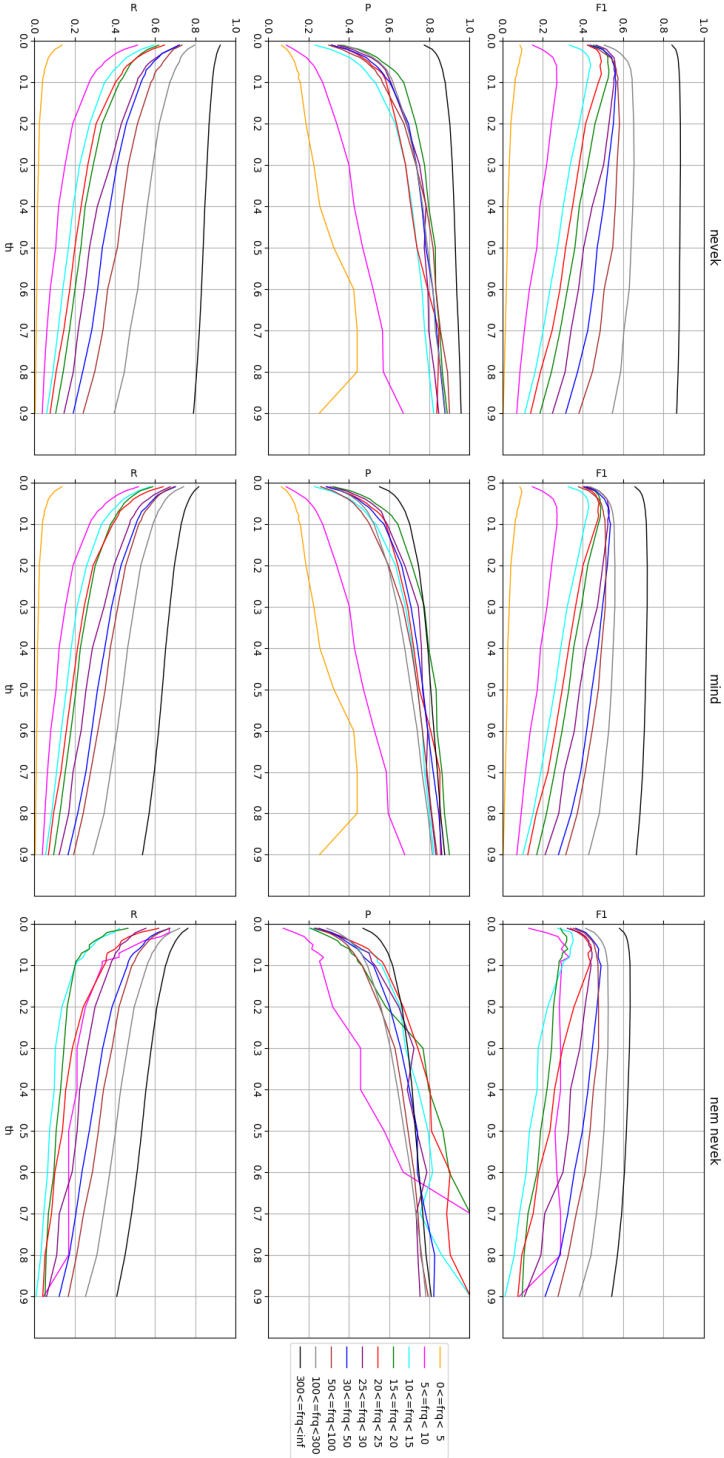
Méréseink eredményét a 2–5. ábrákon mutatjuk be. Az összes címkét figyelembe véve az egyes modellek pontosságának, fedésének és F-mértékének alakulását a vágási küszöb függvényében a 2. ábrán láthatjuk. Láthatóan a hetilapkorpuszon betanított SentencePiece tokenizálóval tokenizált *(heti-sp)* modell nyújtja a legjobb teljesítményt a fedés és az eredő F_1-mérték szerint. A hagyományos tokenizálóval tokenizált szövegen betanított modell *(heti-tok)* teljesítménye gyakorlatilag azonos a tokenizálatlan szövegen betanítottéval *(heti-untok)*. Ugyan ezek pontossága magasabb a *heti-sp* modellénél az alacsonyabb vágási konfidenciaértékeknél, fedésük viszont sokkal rosszabb. Az online korpuszon betanított modell *(online-sp)* mért teljesítménye az eredeti címkék sokkal nagyobb diverzitása miatt elmarad a *heti-sp* modellétől, de a címkék minőségével kapcsolatos szubjektív benyomás a megjelenő szinonim címkék miatt nem rosszabb, sőt a szerkesztőség munkatársai ezt modellt érezték jobbnak. Az online korpusz címkéinek normalizálására létrehoztunk egy eszközt, amelyet jelen kötet egy másik cikkében mutatunk be (Novák és Novák, 2020). A névcímkékre minden modell gyakorlatilag minden szempontból láthatóan jobb teljesítményt nyújt. Az egyetlen kivétel ez alól, hogy hetilapmodelleknél alacsony vágási értékeknél a fedés kicsit jobb a fogalmi címkékre, mint nevekre. F-mértékben a nem SP tokenizált modellek nagyjából a 0,02-es szintnél érik el a maximális teljesítményüket, az SP-modellek ezzel szemben 0,2-nél, de az egész tartományban viszonylag kiegyensúlyozott teljesítményt nyújtanak.

Megmértük a modellek eredő teljesítményét az egyes címkegyakorisági osztályokra külön-külön (3–5. ábrák) is. A *heti-sp* modell (3. ábra) az 5 alatti gyakoriságú nem névcímkék[4] kivételével mindegyik, az *online-sp* modell (4. ábra) pedig gyakorlatilag mindegyik gyakorisági osztályban felmutat valamilyen teljesítményt, bár a fedés a ritkább címkék körében nem túl magas. A 10 alatti gyakoriságú címkéknél a pontosság sem feltétlenül. Általában megfigyelhető, hogy minél gyakoribb egy címke, annál jobb a fedés és a pontosság is, de pl. a *heti-sp* modellnél az 5-9 gyakoriságú fogalmi címkék fedése relatíve kiemelkedik. A hagyományosan tokenizált *heti-tok* modellnél (5. ábra) csak elég alacsony vágási értékeknél kezd megjelenni mérhető fedés. Ennek köszönhetőek a pontossággal kapcsolatos kusza ábrák, hiszen itt a tesztanyagban előforduló viszonylag ritka (30 tanítóminta alatti gyakoriságú) címkék közül (ez a 7684 címkeelőfordulás teszi ki a tesztanyag címkeelőfordulásainak 1/5-ét) csak néhány (konkrétan összesen 40) jelenik meg a 0.05-ös vágási szint felett egyáltalán.

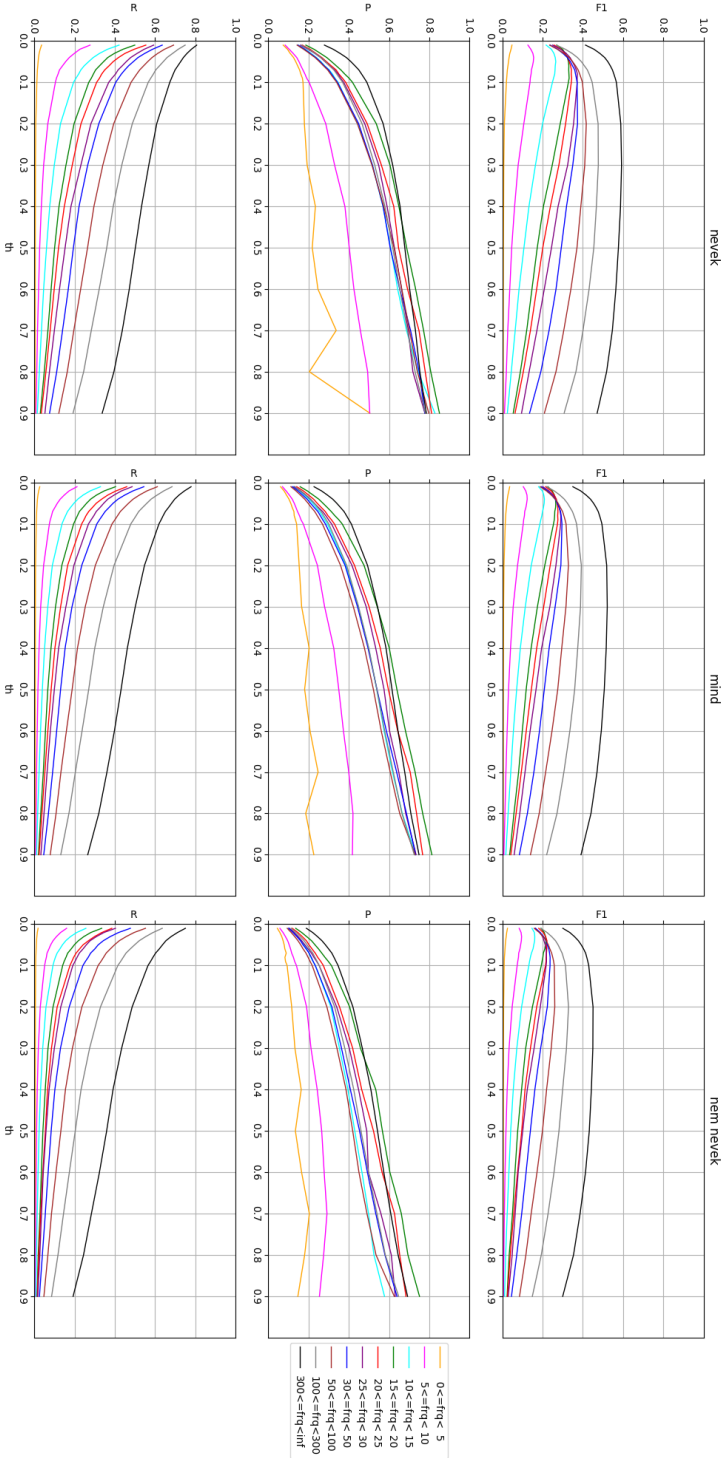
[4] Ilyen címkék nemigen vannak: a teljes tesztkorpuszban összesen 9 címkeelőfordulás tartozik ebbe az osztályba.



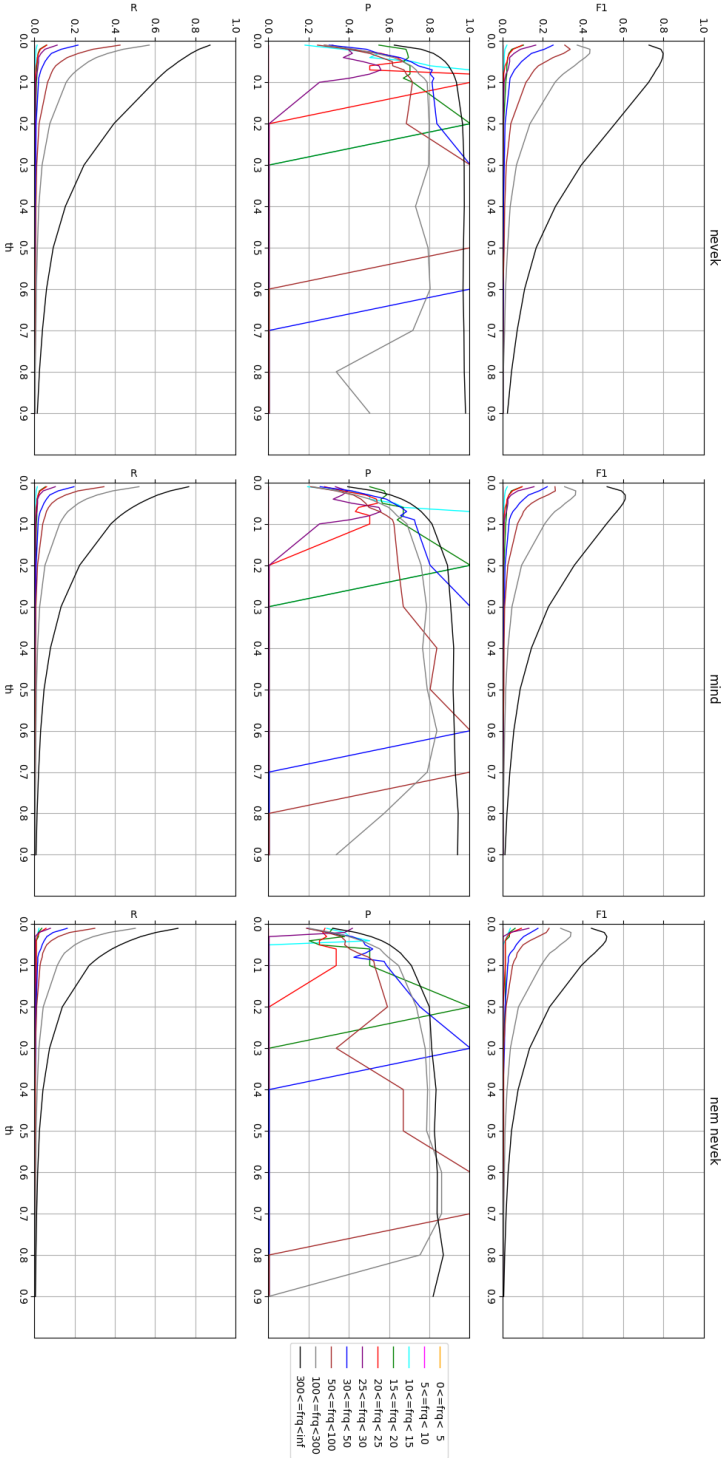
2. ábra: Négy modell teljes címkekészleten nyújtott teljesítménye a vágási küszöb függvényében: P, R, F_1 eredmények



3. ábra: Hetilapmodell, SentencePiece tokenizáló, címkegyakorisági osztályok szerinti P, R, F_1 eredmények a vágási küszöb függvényében



4. ábra: Online modell, SentencePiece tokenizáló, címkegyakorisági osztályok szerinti P, R, F_1 eredmények a vágási küszöb függvényében



5. ábra: Hetilapmodell, hagyományos tokenizálás, címkegyakorisági osztályok szerinti P, R, F_1 eredmények a vágási küszöb függvényében

7. Összegzés

Létrehoztunk egy címkézőrendszert, amellyel sajtószövegek automatikus tematikus címkézését tudjuk megvalósítani. A rendszerhez olyan felhasználói felületet hoztunk létre, amely lehetővé teszi a felhasználó számára a rendszer bizonyos paramétereinek (pl. az ajánlati lista vágását szabályozó konfidenciaszint) beállítását. Ennek segítségével az ajánlott kulcsszólista fedése és pontossága testre szabható. A rendszer segítségével legjobb esetben közel 80%-os pontossággal tudunk tematikus címkéket ajánlani sajtószövegek számára. A fastText osztályozót SentencePiece tokenizálóval kombinálva jelentősen tudtuk növelni a címkézőrendszer fedését, miközben a pontosság csökkenése tolerálható volt, ugyanakkor a modell mérete is töredékére csökkent. Illusztráltuk azt is, hogy a rendszer által ajánlott alacsonyabb konfidenciaértékű címkék, még ha nem szerepeltek is az eredeti címkék között, az esetek nagy részében jól illeszkednek a szöveg témájához.

Köszönetnyilvánítás

Jelen kutatás a Nemzeti Kutatási Fejlesztési és Innovációs Alap által biztosított támogatással az FK 125217 számú projekt keretében az FK 17 pályázati program valamint a Nemzeti Kutatási, Fejlesztési és Innovációs Hivatal Mesterséges Intelligencia Nemzeti Kiválósági Programja támogatásával a 2018-1.2.1NKP-2018-00008 azonosítójú projekt keretében valósult meg.

Hivatkozások

Farkas, R.: Az origo automatikus címkézési projekt tapasztalatai. In: VI. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2009). pp. 84–92. Szegedi Tudományegyetem, Informatikai Tanszékcsoport (2009)

Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. pp. 427–431. ACL, Valencia, Spain (2017)

Kudo, T.: Subword regularization: Improving neural network translation models with multiple subword candidates. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 66–75. ACL, Melbourne, Australia (2018)

Novák, A., Novák, B.: Bu-bor-ék: grafikus címkenormalizáló eszköz. In: XVI. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2020). Szegedi Tudományegyetem, Informatikai Tanszékcsoport (2020)

Szántó, Zs., Vincze, V., Farkas, R.: Magyar nyelvű szó- és karakterszintű szóbeágyazások. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2017). pp. 323–328. Szegedi Tudományegyetem, Informatikai Tanszékcsoport (2017)

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., Le, Q.V.: XLNet: generalized autoregressive pretraining for language understanding. CoRR abs/1906.08237 (2019), http://arxiv.org/abs/1906.08237