Többnyelvű dokumentum nyelvének megállapítása

Pataki Máté[1], Vajna Miklós[1]

[1] MTA SZTAKI Elosztott Rendszerek Osztály
1111 Budapest, Lágymányosi utca 11.
{pataki.mate, vajna.miklos}@sztaki.hu

Kivonat: A cikkben egy olyan algoritmust ismertetünk, amely alkalmas arra, hogy gyorsan és hatékonyan megállapítsa egy szövegről nemcsak annak elsődleges természetes nyelvét, de többnyelvű szöveg esetén a második nyelvet is – mindezt szótár nélkül egy módosított n-gram algoritmus segítségével. Az algoritmus jól működik vegyes nyelvű, akár szótárként felépített, szavanként változó nyelvű dokumentumokon is.

1 Bevezetés

Egy digitális, természetes nyelven íródott dokumentum nyelvének megállapítására számos lehetőség van, és a szakma ezt a problémát nagyrészt megoldottnak tekinti [1][2][3], ugyanakkor a dokumentum nyelvének megállapítása nem mindig egyértelmű feladat.

A leggyakrabban használt algoritmusok igen jól működnek tesztdokumentumokon vagy jó minőségű, gondosan elkészített gyűjteményeken, ha lehet róluk tudni, hogy egy nyelven íródtak. Nekünk azonban szükségünk volt egy olyan algoritmusra, amely internetről letöltött dokumentumokon is jól – gyorsan és megbízhatóan – működik. A KOPI plágiumkereső programunk interneten talált, megbízhatatlan eredetű, gyakran hibás dokumentumokat dolgoz fel, és ennek során lényeges, hogy a dokumentum nyelvét, illetve főbb nyelveit megfelelően ismerje fel, azaz többnyelvű dokumentumok esetében is megbízhatóan működjön.

A jelenleg nyelvfelismerésre használt algoritmusok erre nem voltak képesek magukban, így az egyik algoritmust úgy módosítottuk, hogy amennyiben egy dokumentumban nagyobb mennyiségben található más nyelvű szöveg, akkor azt jelezze, és így a plágiumkereső rendszer ezt mint többnyelvű dokumentumot tudja kezelni.

Az algoritmussal szemben az alábbi elvárásokat támasztottuk:

1. Jelezze, ha a dokumentum több nyelven íródott, és nevezze meg a nyelveket
2. Az algoritmus gyors legyen
3. A szöveget csak egyszer kelljen végigolvasni
4. Ne szótár alapú legyen (kódolási és betanítási problémák miatt)

A legegyszerűbb megoldásnak az n-gram algoritmus tűnt [1][4], mivel ezen algoritmust használva csak egyszer kell végigolvasni a dokumentumot és az n-gram sta-

tisztikákból meg lehet állapítani, hogy a dokumentum milyen nyelven íródott, és – ha vannak megfelelő mintáink – még a kódolását is meg tudja határozni.

Az n-gram viszont nem teljesíti az első feltételt, miszerint a több nyelven íródott dokumentumokat is fel kell ismernie. Ugyan elméletileg elképzelhető lenne, hogy a dokumentumot szakaszokra osztjuk, és szakaszonként állapítjuk meg a dokumentum nyelvét, de ez a megoldás sajnos két esetben is hibás eredményre vezet. Gyakran találkoztunk olyan dokumentummal, amelyik úgy volt felépítve, mint egy szótár, azaz a két nyelv nem szakaszonként, hanem mondatonként – sőt egyes esetekben szavanként – váltakozott. A másik probléma akkor jelentkezett, amikor a dokumentum – például egy korábbi hibás konverzió miatt – tartalmazott HTML- vagy XML-elemeket, amelyek miatt rövid dokumentumok esetében hibásan angol nyelvűnek találta az algoritmus azokat.

Ezek kiküszöbölésére kezdtük el továbbfejleszteni az n-gram algoritmust, amely alapból csak arra alkalmas, hogy a dokumentumban leggyakrabban használt nyelvet megállapítsa, de a második leggyakoribb nyelv már nem a második a listában. Ennek oka, hogy a nyelvek hasonlítanak egymásra, és például egy nagyrészt olasz nyelvű dokumentum esetében a spanyol nyelv akkor is nagyobb értéket kap, mint a magyar, ha a dokumentum egy része magyar nyelven íródott.

Az új algoritmusunkba ezért beépítettünk egy nyelvek közötti hasonlósági metrikát, amelyet a hamis találatok értékének a csökkentésére használunk. A metrika segítségével meg lehet állapítani, hogy a második, harmadik... találatok valódiak-e, vagy csak két nyelv hasonlóságából fakadnak.

2 Az eredeti algoritmus

Az n-gram algoritmus működése igen egyszerű, legenerálja egy nyelvnek a leggyakoribb „betű n-gramjait”, azaz a például 1, 2, 3 betű hosszú részeit a szövegnek, majd ezeket az előfordulási gyakoriságuk szerint teszi sorba. A magyar nyelvben ez a 100 leggyakoribb n-gram az általunk használt tesztszövegben (_ a szóköz jele):

1. _	17. y	33. s_	49. er
2. e	18. _a	34. _m	50. f
3. a	19. b	35. _a_	51. ek
4. t	20. d	36. en	52. te
5. s	21. a_	37. ö	53. és
6. l	22. v	38. n_	54. _s
7. n	23. t_	39. _k	55. al
8. k	24. sz	40. j	56. ta
9. i	25. el	41. ._	57. í
10. r	26. ,	42. i_	58. _h
11. z	27. ,_	43. eg	59. _t
12. o	28. h	44. p	60. an
13. á	29. k_	45. _e	61. ze
14. é	30. .	46. u	62. me
15. g	31. et	47. le	63. at
16. m	32. gy	48. ó	64. l_

65. es	74. _é	83. ne	92. _A
66. ő	75. ny	84. os	93. _sz
67. y_	76. tá	85. ál	94. is
68. z_	77. c	86. _f	95. ve
69. tt	78. re	87. az	96. gy_
70. ke	79. to	88. zt	97. ít
71. _v	80. A	89. ár	98. _b
72. ás	81. e_	90. _n	99. ra
73. ak	82. ü	91. ko	100. or

Két szöveg összehasonlítása úgy történik, hogy a két n-gram listán összeadjuk az azonos n-gramok helyezéseinek a különbségét, és ez adja a két dokumentum közötti hasonlóság mértékét. Két azonos nyelven írt dokumentum között alig, míg különböző nyelvek között szignifikáns lesz a különbség. Ezért használható ez az algoritmus a dokumentum nyelvének megállapítására.

Példának nézzük meg az angol nyelvű példadokumentumunk első 10 n-gramját, és hasonlítsuk össze a magyarral.

1. _ (1-1)
2. e (2-2)
3. t (3-4)
4. o (4-12)
5. n (5-7)
6. i (6-9)
7. a (7-3)
8. s (8-5)
9. r (9-10)
10. h (10-28)

Az eredmény 0+0+1+8+2+3+4+3+1+18 = 40. Ez a különbség egyre nagyobb lesz, ahogy lejjebb megyünk a listában. Mivel nem lehet végtelen hosszú listát készíteni, így azokat az n-gramokat, amelyek az egyik listában szerepelnek, de a másikban nem, úgy vesszük figyelembe, mintha a lista utolsó helyén álltak volna. Mi egy 400-as listával dolgoztunk, azaz az első 400 n-gramot tároltuk el minden nyelvhez.

Ennek megfelelően a két nyelv elméleti minimális távolsága 0, maximális távolsága (r_{max}) pedig 400^2 azaz 160 000. Ebből a százalékos hasonlóságot a

$$h_{százalékos} = \left(r_{max} - r\right)/\left(r_{max}/100\right)$$

összefüggéssel kapjuk.

Példának nézzük meg, hogy mekkora hasonlóságot mutatnak különböző nyelvű dokumentumok a mintadokumentumainkhoz képest. Az egyszerűbb olvashatóság érdekében $h_{százalékos}$ értékekkel számolva a különböző nyelvű **Szeged Wikipédia-szócikkekre** [5][6][7][8][9].

A **magyar** nyelvű szócikk esetén az alábbi eredményt kapjuk, az első 5 találatot kérve:

1. magyar: 35.49
2. breton: 27.70
3. szlovák: 27.42
4. eszperantó: 26.98
5. közép-frízi: 26.79

Az **angol** nyelvű szócikk esetén az alábbi eredményt kapjuk:

1. angol: 44.37
2. skót: 35.67
3. romans: 35.34
4. német: 33.74
5. román: 33.73

A **német** nyelvű szócikk esetén az alábbi eredményt kapjuk:

1. német: 57.13
2. holland: 38.15
3. közép-fríz: 37.71
4. dán: 37.48
5. fríz: 36.58

Az **olasz** nyelvű szócikk esetén az alábbi eredményt kapjuk:

1. olasz: 35.21
2. román: 33.95
3. katalán: 33.46
4. spanyol: 32.18
5. romans: 31.78

Jól látható az eredményekből, hogy a barátságos nyelvek esetében magas hasonlóságot mutat a dokumentum a rokon nyelvekre, azaz egy olasz nyelvű dokumentum majdnem ugyanannyi pontot kap az olaszra, mint a spanyolra.

Most nézzük meg, hogy **kétnyelvű, 50-50 százalékban kevert dokumentumok**ra mit kapunk.

Egy **magyar-angol** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. angol: 40.80
2. magyar: 39.45
3. skót: 38.41
4. afrikaans: 34.69
5. közép-fríz: 34.19

Egy **magyar-olasz** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. olasz: 49.56
2. romans: 45.25
3. katalán: 41.60
4. latin: 41.26
5. román: 41.18

…

10. magyar: 38.02

Egy **magyar-francia** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. francia: 38.16
2. katalán: 36.74
3. eszperantó: 34.26
4. spanyol: 34.08
5. romans: 33.71

…

7. magyar: 33.2

Egy **angol-német** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. német: 53.47
2. angol: 44.14
3. fríz: 40.98
4. közép-fríz: 40.61
5. holland: 40.08

Látható, hogy a magyar-olasz, ill. magyar-francia kevert szövegben a magyar nyelv bele se került az első 5 találatba.

Végül nézzük meg, hogy egy háromnyelvű, harmadolt arányban kevert dokumentumra mit kapunk.

Egy **magyar-angol-olasz** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. angol: 46.55
2. olasz: 44.55
3. romans: 43.58
4. katalán: 42.41
5. román: 41.11

…

10. magyar: 38.26

Láthatjuk, hogy a háromnyelvű szövegben sem kerül be az első öt helyre a magyar nyelv.

3 Az új algoritmus

Mint láttuk, bizonyos nyelvek hasonlítanak egymásra az n-gram algoritmus szempontjából, így egy többnyelvű dokumentum esetén a második helyen nem minden esetben a dokumentum második nyelvét találjuk, ráadásul az se derül ki, hogy a második nyelv azért került oda, mert valóban szerepel a dokumentumban, vagy azért, mert hasonlít az első nyelvre. Ezért az új algoritmusunkban elkezdtük kiszámolni a **nyelvek közötti hasonlóság**ot, méghozzá a nyelvfelismeréshez használt n-gram minták közötti hasonlóságot. A távolságok tipikus értékeire nézzünk néhány esetet.

A **magyar** nyelvhez legközelebb álló nyelvek távolság-értékei:

1. breton: 104 541
2. közép-fríz: 104 751
3. svéd: 106 068

4. eszperantó: 106 469
5. afrikaans: 106 515

Az **angol** nyelvhez legközelebb állók:

1. skót: 85 793
2. francia: 88 953
3. katalán: 89 818
4. latin: 90 276
5. romans: 92 936

Végül az **olasz** nyelvhez legközelebb állók:

1. romans: 79 461
2. román: 85 232
3. katalán: 85 621
4. spanyol: 86 138
5. latin: 86 247

Számos algoritmussal próbálkoztunk, melyek közül az alább leírt bizonyult a legmegbízhatóbbnak.

Egy D dokumentumra kapott százalékos hasonlóságaink (hszázalékos), a százalékos hasonlóság mértékének növekvő sorrendjében legyen: h1, h2, h3 stb., a nyelveket jelölje L1, L2, L3, azaz a h1 a D dokumentum hasonlóságát mutatja az L1 nyelvű mintánkkal százalékban. A nyelvek közötti százalékos hasonlóságot pedig jelöljük hL1L2-vel. hi' legyen az új algoritmus által az Li nyelvre adott érték.

$$h_i' = h_i \quad ha \quad i = 1$$

$$h_i' = h_i - \frac{\sum_{k=1}^{i-1} h_k \times h_{LiLk}}{\sum_{k=1}^{i-1} h_i} \quad ha \quad i > 1$$

Az algoritmus tulajdonképpen minden nyelv valószínűségét csökkenti az előtte megtalált nyelvek valószínűségével, így kompenzálva a nyelvek közötti hasonlóságból adódó torzulást. Példának nézzük meg, hogy mekkora hasonlóságot mutatnak különböző nyelvű dokumentumok a mintadokumentumainkhoz képest ezzel az új algoritmussal számolva.

Egy **magyar** nyelvű dokumentum (Szeged Wikipédia-szócikke) esetén az alábbi eredményt kapjuk, az első 5 találatot kérve:

1. magyar: 35.49
2. kínai: 2.09
3. japán (euc jp): 1.81
4. koreai: 1.70
5. japán (shift jis): 1.58

Egy **angol** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. angol: 44.21
2. nepáli: 3.84
3. kínai: 2.53
4. vietnami: 2.08
5. japán: 1.14

Egy **német** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. német: 57.13
2. kínai: 2.55
3. japán (shift jis): 2.19
4. japán (euc jp): 1.93
5. nepáli: 1.27

Egy **olasz** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. olasz: 35.21
2. kínai: 1.07
3. perzsa: 0.68
4. japán: 0.57
5. jiddis: 0.55

Jól látható az eredményekből, hogy a barátságos nyelvek esetében a nyelvek hasonlóságából adódó hamis többletpontok kiszűrésre kerültek, azaz egy olasz nyelvű dokumentumnál a spanyol nyelv már meg se jelenik az első öt találatban. Most nézzük meg, hogy a **kétnyelvű, 50-50 százalékban kevert dokumentumok**ra mit kapunk.

Egy **magyar-angol** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. angol: 40.80
2. magyar: 9.40
3. thai: 1.54
4. armeniai: 1.39
5. koreai: 1.37

Egy **magyar-olasz** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. olasz: 49.56
2. magyar: 7.44
3. walesi: 2.31
4. breton: 1.92
5. ír: 1.68

Egy **magyar-francia** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. francia: 38.16
2. magyar: 2.11
3. thai: 1.42
4. koreai: 1.16
5. kínai: 0.70

Egy **angol-német** nyelvű dokumentum esetén az alábbi eredményt kapjuk:

1. német: 53.47
2. angol: 7.79
3. walesi: 2.08

4. fríz: 1.48
5. nepáli: 1.44

Látható például, hogy a magyar-olasz kevert szövegben a magyar nyelv immár a 2. helyre került, a korábbi – eredeti algoritmus által megadott – 10. helyről.

A kétnyelvű dokumentumok esetében nem mindegy, hogy a nyelvek milyen arányban keverednek, érthető módon egy bizonyos arány felett az egyik nyelv n-gramjai elnyomják a másikét. Ezt egy **angol-magyar** dokumentumsorozat segítségével nézzük meg. Az egyes részek aránya a 9 dokumentum során a 10% angol, 90% magyar összetételről 90% angol és 10% magyar összetételre változott:

10% angol, 90% magyar: 1. magyar: 38.01 2. koreai: 1.53 3. thai: 1.20 4. japán (euc): 1.14 5. japán (shift): 1.09	40% angol, 60% magyar: 1. angol: 37.62 2. magyar: 5.41 3. japán (euc): 1.47 4. thai: 1.46 5. japán (shift): 1.45	70% angol, 30% magyar: 1. angol: 44.92 2. vietnámi: 1.74 3. mingo: 1.67 4. kínai: 1.46 5. armén: 1.36
20% angol, 80% magyar: 1. magyar: 37.93 2. thai: 1.18 3. koreai: 1.17 4. japán: 1.16 5. armén: 1.11	50% angol, 50% magyar: 1. angol: 40.93 2. magyar: 5.30 3. thai: 1.49 4. japán (shift): 1.47 5. japán (euc): 1.37	80% angol, 20% magyar: 1. angol: 46.56 2. vietnámi: 2.07 3. mingo: 2.00 4. japán: 1.47 5. walesi: 1.43
30% angol, 70% magyar: 1. magyar: 37.47 2. angol: 4.91 3. thai: 1.22 4. armén: 1.18 5. japán: 1.16	60% angol, 40% magyar: 1. angol: 41.66 2. magyar: 3.43 3. kínai: 1.50 4. vietnámi: 1.48 5. mingo: 1.45	90% angol, 10% magyar: 1. angol: 48.1 2. vietnámi: 1.51 3. nepáli: 1.40 4. thai: 1.05 5. kínai: 1.05

A fenti táblázat csak egy példa, de a többi nyelvpárra is hasonló eredményeket kaptunk. Látható, hogy az algoritmus 30% körül kezd el hibázni, azaz akkor találja meg megbízhatóan a második nyelvet, ha az a szöveg több mint 30%-át teszi ki.

Hasonló eredményt kapunk egy **háromnyelvű**, harmadolt arányban kevert, **magyar-angol-olasz** nyelvű dokumentum esetén is:

1. angol: 46.55
2. magyar: 7.59
3. olasz: 6.18
4. breton: 3.11
5. skót: 2.85

Láthatjuk, hogy a háromnyelvű szövegben az első három helyen szerepelnek a valós nyelvek, de azért itt el kell mondani, hogy ez csak az egyenlő arányban kevert háromnyelvű dokumentumok esetén működik jól. Ha ez az arány eltolódik, akkor gyorsan kieshet egy-egy nyelv. Tapasztalatunk szerint az új algoritmus három nyelvet már nem talál meg megbízhatóan, így ilyen dokumentumok tömeges előfordulása esetén más algoritmust ajánlott választani.

5 Konklúzió

Ahhoz, hogy megállapítsuk, egy dokumentum egy vagy több nyelven íródott-e, kell választanunk egy olyan értéket, ami felett azt mondjuk, hogy a második nyelv is releváns, azaz a dokumentum többnyelvű. Ezt az értéket a tesztek alapján 4-nek választottuk, azaz 4-es érték felett jelezzük csak ki a nyelveket. Ez az érték a felhasználási igényeknek megfelelően választható. Akkor érdemes valamivel alacsonyabbra állítani, ha mindenképp észre szeretnénk venni, ha a dokumentum kétnyelvű, ha pedig csak igazán nagy idegen nyelvű részek érdekelnek, és nem okoz gondot a hibásan egynyelvűnek talált dokumentum, akkor állíthatjuk akár magasabbra is.

Ezzel a paraméterrel az algoritmust részletesen teszteltük a plágiumkeresőnkbe feltöltött dokumentumokon, és a vele szemben támasztott igényeknek messzemenőkig megfelelőnek találtuk. Ki tudtuk szűrni vele a rosszul konvertált és többnyelvű dokumentumok több mint 90%-át. A tesztek befejezése után az új algoritmust beépítettük a KOPI Plágiumkereső rendszerbe, ahol a korábbi, kevésbé pontos eredményt adó algoritmust váltotta ki.

Bibliográfia

1. Cavnar, W. B.; Trenkle, J. M.: N-Gram-Based Text Categorization. Proceedings of Third Annual Symposium on Document Analysis and Information Retrieval. UNLV Publications/Reprographics, Las Vegas, NV, (1994) 161-175
2. Řehůřek, R.; Kolkus, M.: Language Identification on the Web: Extending the Dictionary Method. In: 10th International Conference on Intelligent Text Processing and Computational Linguistics (2009)
3. Benedetto, D.; Caglioti, E.; Loreto. V.: Language trees and zipping. Physical Review Letters Vol. 88, No. 4 (2002)
4. Dunning, T.: Statistical Identification of Language. Technical Report MCCS 94-273, New Mexico State University (1994)
5. Wikipedia: Szeged szócikk magyar nyelven, http://hu.wikipedia.org/wiki/Szeged (2011)
6. Wikipedia: Szeged szócikk angol nyelven, http://en.wikipedia.org/wiki/Szeged (2011)
7. Wikipedia: Szeged szócikk német nyelven, http://de.wikipedia.org/wiki/Szeged (2011)
8. Wikipedia: Szeged szócikk olasz nyelven, http://it.wikipedia.org/wiki/Seghedino (2011)
9. Wikipedia: Szeged szócikk francia nyelven, http://fr.wikipedia.org/wiki/Szeged (2011)