Karakter *n*-gramm alapú nyelvfelismerés

Nyelvfelismerés és Zipf törvénye

A nyelvfelismerés feladata eldönteni, hogy egy adott szöveget milyen nyelven írtak. A megoldással szemben elvárjuk, hogy

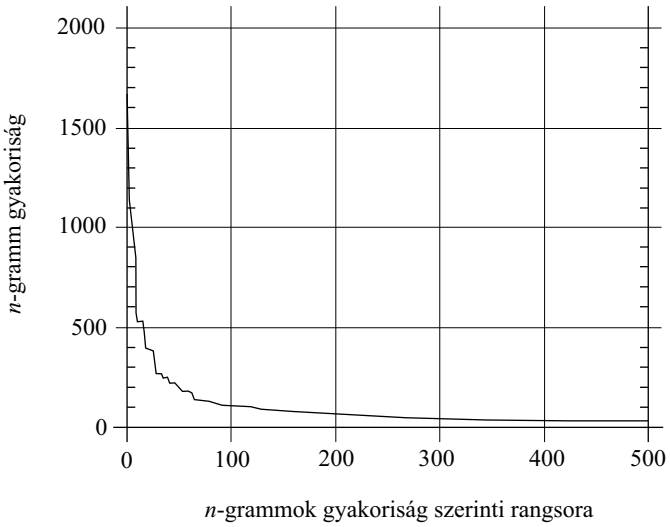
- hibatűrő legyen, azaz megbízhatóan működjön olyan, esetenként sok gépelési és helyesírási hibát tartalmazó szövegekre is, mint az internetes dokumentumok,
- a legtöbb karakter alapú nyelvet felismerje,
- és mindezt minél jobb hatékonysággal valósítsa meg.

A Canvar és Trenkle által javasolt nyelvfelismerő megoldás (Canvar & Trenkle, 1994) a dokumentumok karakter *n*-gramm alapú indexelését használja fel. Az *n*-grammok a dokumentum *n* hosszúságú részsztringjei, amelyeket tokenekre — azaz szóközökkel szeparált stringekre — értelmezünk. A tokeneket az elején pontosan 1, végén pedig maximum $n-1$ szóközzel feltöltjük az *n*-grammok képzésénél, így a *szöveg* szó *n*-grammjai $n = 2,3,4$ értékekre az alábbiak lesznek

- bigrammok ($n = 2$): ␣s, sz, zö, öv, ve, eg, g␣;
- trigrammok ($n = 3$): ␣sz, szö, zöv, öve, veg, eg␣, g␣␣;
- 4-grammok ($n = 4$): ␣szö, szöv, zöve, öveg, veg␣, eg␣␣, g␣␣␣;

Ezzel a módszerrel egy ℓ hosszú sztring $\ell+1$ *n*-grammot (bi-, tri, 4-grammot stb.) generál.

Minden természetes nyelvre jellemző, hogy egyes szavak gyakoribbak más szavaknál. Ezt fogalmazza meg a relatív szógyakoriságokat leíró Zipf törvénye, amely kimondja, hogy az *n*-edik leggyakoribb szó egy természetes nyelvű szövegben *n*-nel fordítottan arányos gyakorisággal fordul elő. Ebből következik, hogy vannak olyan szavak, amelyek dominánsak, azaz a szöveg egyéb jellemzőitől (téma, stílus, hossz stb.) függetlenül gyakrabban fordulnak elő más szavaknál. Az átmenet a gyakori és a ritka szavak között folytonos, azaz ugrásmentes. Ez a jellegzetesség nemcsak a szavakra, hanem az *n*-grammokra is fennáll (ld. az 1. ábrát). Az egyes nyelvekre jellemző, hogy melyek a leggyakrabban előforduló szavaik, *n*-grammjaik. A nyelvfelismerő algoritmus a nyelvek e tulajdonságát használja fel.



1. ábra. N-grammok gyakoriságának és gyakoriság szerinti rangsorának összefüggése

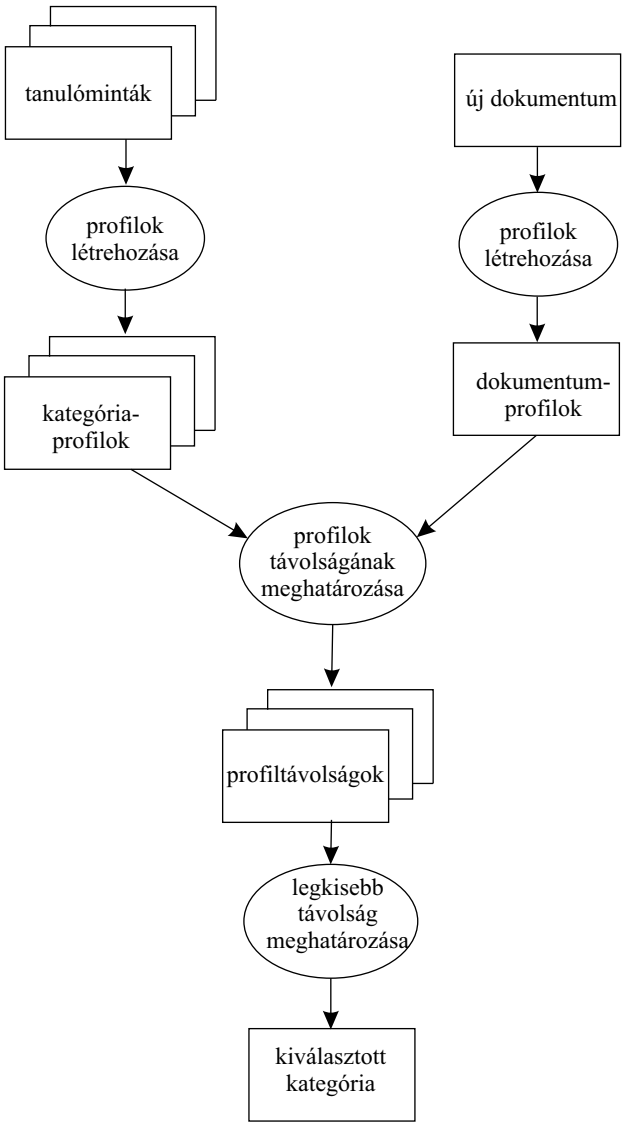
A módszer működése

A módszer működését illusztrálja a 2. folyamatábra. A profilok az adott szöveg leggyakoribb *n*-grammjait állítják elő. A profilok generálása az alábbi lépésekből épül fel:

- A bemeneti szöveg tokenizálása és a felesleges karakterek eldobása (számjegyek, írásjelek).
- Minden tokenből *n*-grammok generálása, $n = 1, \ldots, 5$ értékekre a fent leírt módon.
- Az *n*-grammok számát hash-függvény segítségével tároljuk. Ha egy adott *n*-gramm már létezik, akkor a számlálót növeljük, különben új értéket veszünk fel.
- Végül az *n*-grammokat előfordulások szerint csökkenő sorrendbe állítjuk, és a leggyakoribb *N*-ből képezzük a dokumentum profilját.

A profilok alapján az alábbi megfigyeléseket lehet tenni.

- Az azonos nyelvű dokumentumok profilja $N = 300$ esetén nagyon hasonló egymáshoz, függetlenül a dokumentum egyéb jellemzőitől. Ugyanakkor kü-



2. ábra. Az *n*-gramm alapú osztályozó folyamatábrája

lönböző nyelvű szövegek profilja, még ha a dokumentum témája és stílusa azonos is, nagyon eltérő.

- A leggyakoribb *n*-grammok az unigrammok, amelyek a nyelv által használt ábécé elemeinek gyakoriságát adják meg. Ezután a gyakori szavak részsztringjei következnek, majd az egyre hosszabb nyelvspecifikus *n*-grammok.
- $N = 300$ után jelennek meg a gyakorisági listán a témaspecifikus *n*-grammok. A konkrét érték függ a nyelvfelismerésre alkalmazott tanítókorpusz jellegétől is. A 300-as érték a rövid szövegekből álló korpuszra érvényes, hosszabb szövegek esetén a témaspecifikus *n*-grammok a rangsorban hátrébb helyezkednek el.

A profilok összehasonlítását pozícióindex szerint végezzük. Legyen $\mathrm{pos}_i(k)$ a k *n*-gramm pozíciója az i profilban, ekkor a $\mathbf{d}_i$ dokumentumprofil távolságát a $\mathbf{c}_j$ kategóriaprofiltól a

$$d(\mathbf{d}_i, \mathbf{c}_j) = \sum_{k=1}^{N} \begin{cases} (\mathrm{pos}_i(k) - \mathrm{pos}_j(k)), & \text{ha } k \in \mathrm{pos}_j \\ N, & \text{ha } k \notin \mathrm{pos}_j \end{cases}$$

összeg határozza meg. A dokumentumot ezután ahhoz a kategóriához soroljuk be, amelynek profiljához a legközelebb van (ld. még a Rocchio-algoritmust).

Ez az egyszerű algoritmus nagyon jó hatékonysággal működik nyelvfelismerésre. A szótár alapú nyelvfelismerésnél előnyösebb, mivel nincs szükség az esetenként nehezen hozzáférhető, nagyméretű szótárak kezelésére, és tanítóadatként csupán néhány száz szavas dokumentumok is elegendőek. A rendszert 14 országból származó 3478 hírcsoportüzeneten tanították be és tesztelték. A nyelvek között csupán dialektusukban különbözők is voltak (pl. angol és ausztrál, brazil és portugál), de a megfelelő üzeneteket nem vonták össze, hanem eltérő nyelvű szövegként kezelték őket. A tesztrendszer ennek ellenére $N = 300$-as értékre a 300 karakternél(!) nem hosszabb szövegek esetén átlagosan 98,6%-os pontosságot ért el, 300 karakternél hosszabb szövegekre pedig átlagosan 99,8%-os pontosságot. A legnehezebben a brazil és portugál eredetű szövegeket tudta megkülönböztetni a rendszer, a brazil szövegekre 95,7-es pontosságnál jobbat nem tudtak elérni. $N = 400$-ra is hasonló értékeket adott a rendszer.

Az itt ismertetett módszert nemcsak nyelvfelismerésre, hanem tematikus osztályozásra is lehet alkalmazni, de ekkor lényegesen rosszabb eredményt produkál.

Forrás:
W. B. Canvar and J. M. Trenkle. N-gram-based text categorization. In *Proc. of SDAIR-94, 3rd Annual Symp. on Document Analysis and Information Retrieval*, pp. 161–175, Las Vegas, USA, 1994.