**Érzelmek felismerése Twitter-üzenetekben**

# Feladat

A hallgató feladata olyan mesterséges intelligencia készítése, amely képes pozitív és negatív érzelmeket felismerni Twitter-üzenetekben. Az üzenetek alkalmas reprezentációja nem része a feladatnak, ezeket a hallgató megkapja; a teljesség kedvéért azonban összegezzük, hogyan lehet az algoritmus számára is könnyen kezelhető formátumra hozni őket. Ehhez először meghatározzuk az üzenetben lévő szavak szófaját (part-of-speech tagging), majd ezt felhasználva előállítjuk a szavak szótári alakját (lemmatizáció). A @-cal kezdődő szavakat nem vesszük figyelembe. Minden szó kap egy azonosítót, majd minden üzenetet a benne lévő szavak azonosítóival reprezentálunk:

fail maths i knew it i was just too slow

→ [(fail*,*JJ)*,*(maths*,*NNS)*,*(i*,*VBP)*,*(knew*,*VBD)*,*(it*,*PRP)*,*(i*,*NN)*,*(was*,*VBD)*,*(just*,*RB)*,*(too*,*RB)*,*(slow*,*JJ)] → [fail*,*math*,*i*,*know*,*it*,*i*,*be*,*just*,*too*,*slow]

→ [87546*,*25854*,*177452*,*55538*,*184147*,*177452*,*107857*,*190578*,*110607*,*23550]

**Példa.** Néhány üzenet az adatbázisból:

@blzl Melted Cheese and Chips ! Aww... Sounds Yummy !

@cliquedecamwa i hate that too. the site won’t even load

Az ezekhez tartozó reprezentációk, amelyek a bemenetet fogják képezni:

7789 13570 65745 181474 75021 164183 8299

187531 8708 167996 68470 186092 91580 188225 187296 154979

Minden bemenethez tartozik egy (ember által meghatározott) címke is, amely a pozitív vagy negatív érzelmi töltést reprezentálja:

1

0

**A feladat során a tanító dokumentumok és címkék felhasználásával egy klasszifikációs modellt kell tanulni (pl. naiv Bayes), majd ezt felhasználni a teszt dokumentumok címkéinek jóslására.**

## Bemenet

A hallgató a standard inputon kapja meg a tanító dokumentumok reprezentációit, a hozzájuk tartozó címkéket, valamint a teszt dokumentumok reprezentációit. A teljes bemenet a következőképpen épül fel:

1. A bemenet első 80000 sora egy-egy dokumentumot tartalmaz, benne a szavak azonosítóival, \t karakterrel elválasztva (tanító minták).
2. Ezt követi a hozzájuk tartozó 80000 tanító címke (soronként 1 db 0 vagy 1).
3. Ezt követi 20000 dokumentum, amelyekhez a címkéket meg kell határozni (teszt minták).

## Kimenet

A megoldás a teszt mintákra adott predikciókat tartalmazza, soronként egyet. A megoldást a standard outputra kell kiírni, a sor szeparátor a \n karakter.

**Példa.** A kimenet 20000 sorból áll:

1

0

0

1 ...

# Fontos tudnivalók

* A kódot Java-ban kell írni, nem tartalmazhat ékezetes vagy nem ASCII[0:127] karaktert. A beadott forráskódnak tartalmaznia kell egy Main osztályt, azon belül egy main() függvényt. Külső csomagokat nem lehet használni.
* Az üzenetek lehetnek üresek, valamint a teszthalmazban lehetnek olyan szavak, amelyek korábban nem fordultak elő.

# Értékelés

A címkéket 60%-ban eltaláló megoldás 1 pontot ér, a 80% fölött teljesítő a maximális 12 pontot. E két végpont között az értékelés lineáris (de csak egész pontot lehet kapni).