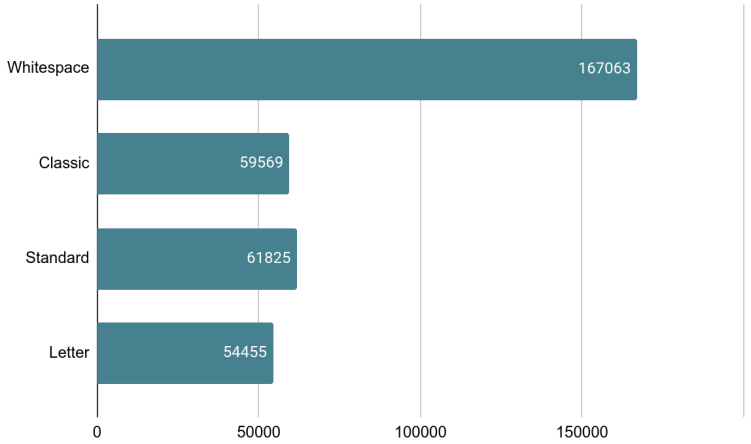
Index pipeline

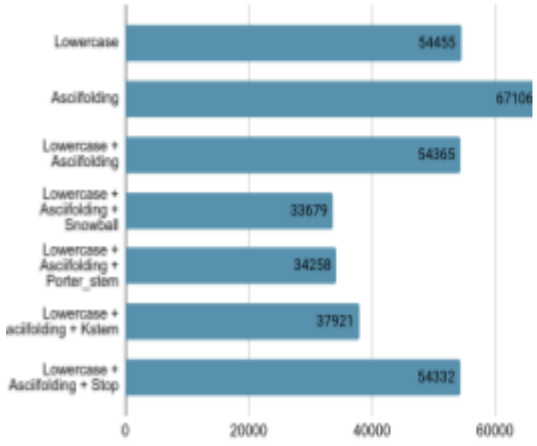
A la primera part d'aquesta pràctica aprenem com fer servir les diferents opcions del fitxer *IndexFilesPreprocess.py*, per indexar documents. Més específicament ens concentrem en els flags –token i –filter per veure com afecten cadascuna de les opcions que tenim disponibles a cada token.
Comencem per provar només el flag-token amb els diferents tokenizers sobre el conjunt de dades *novels*.
Obtenim els resultats següents:



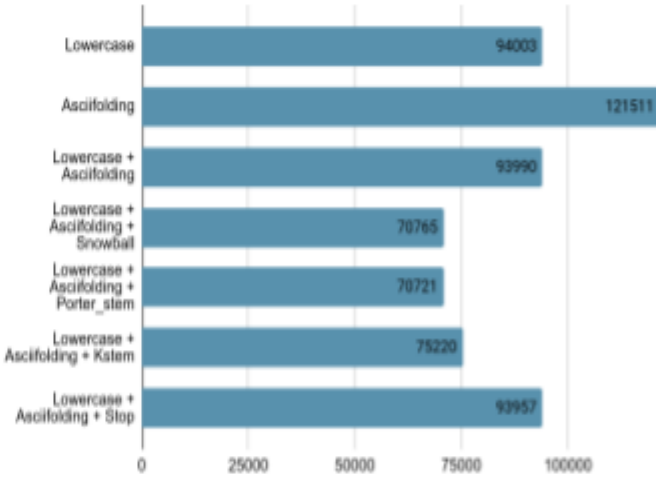
A la gràfica tenim a l'eix y les diferents opcions de tokenizers i a l'eix x el nombre de tokens que ens queden després d'aplicar cada opció sobre el conjunt de dades d'entrada.

Podem veure clarament que l'opció més efectiva de totes és *Letter*, ja que és la més agressiva. Per contra l'opció *Whitespace* és la que més tokens ens deixa respecte a la totalitat de tokens inicial, lògicament, pel fet que només elimina els espais en blanc. L'opció *Standard* elimina més signes de puntuació i el *Classic* s'usa més per a l'idioma anglès. *Letter* divideix el text en termes cada vegada que troba un caràcter que no és una lletra, es fa servir més per a qualsevol llenguatge europeu.

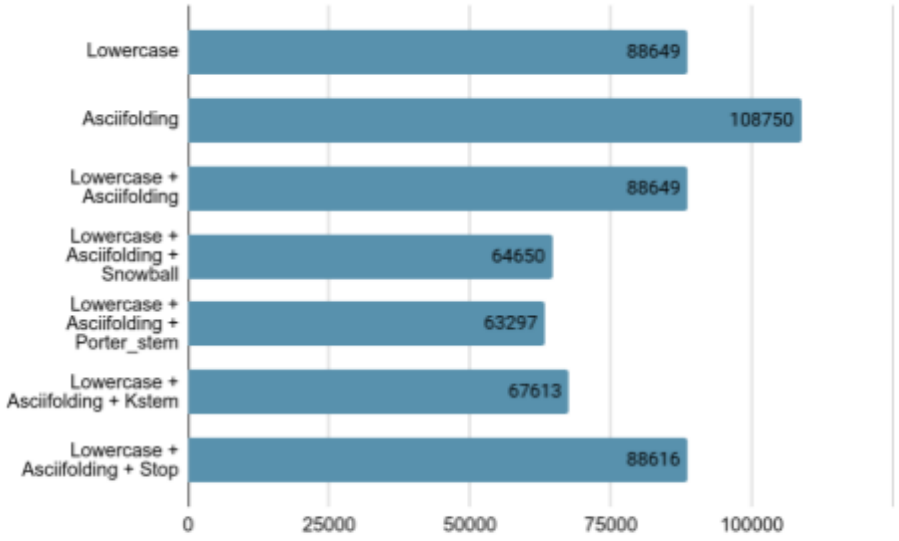
Per continuar amb l'estudi farem servir el tokenizer *Letter* per les raons enunciades anteriorment.



novels



20_newsgroups

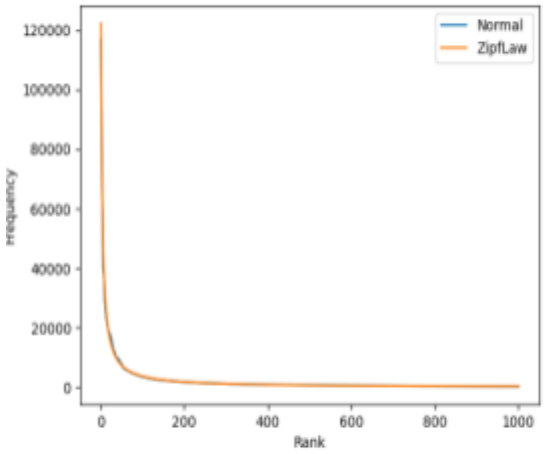


arxiv_abs

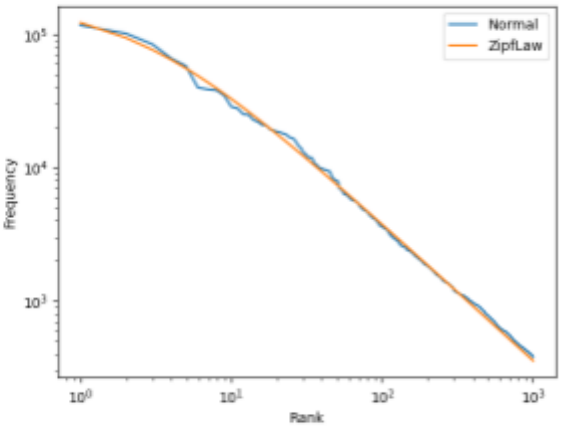
Hem provat diferents combinacions de filtres. De les gràfiques podem veure que al *novels* el més efectiu és el filtre *snowball* i a les altres la diferència entre l'*snowball* i el *porter stem* és gairebé insignificant. Cal tenir en compte l'ordre d'aplicació dels filtres i la utilització, ja que podem arribar a eliminar paraules que ens interessen, com acrònims o números, o bé empobrir el contingut lèxic a causa de l'eliminació o transformació de certes paraules,
com ara els accents o paraules amb un significat molt diferent que puguin compartir l'arrel. Per aquestes raons decidim fer servir a la segona part de la pràctica el filtre *snowball*.

Zipf's Law

Tot seguit estudiem si es compleix la llei de Zipf per al conjunt de dades novels.

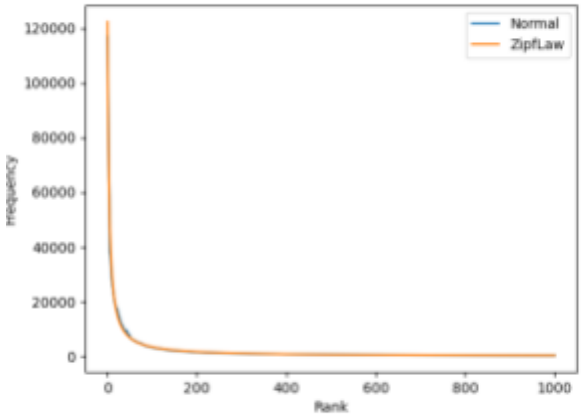


Snowball No log

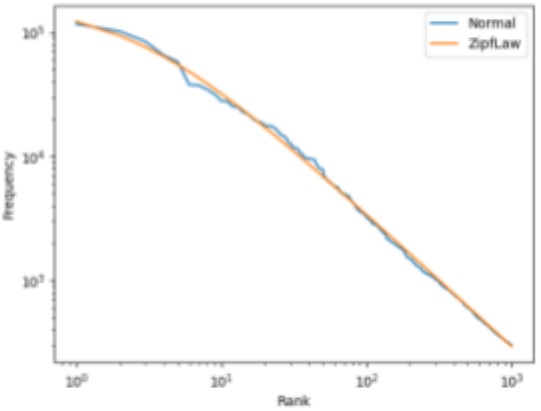


Snowball Log

Com veiem de les gràfiques, la llei de Zipf se segueix complint quan fem servir el flag -token amb *letter* i el -filtre amb *Lowercase + Asciifolding + Snowball*.



novels no log



novels log

És difícil dir si empitjora o millora respecte a la versió de la pràctica anterior. Sembla que millora una mica si observem atentament els gràfics, però el resultat s'assembla molt.

Càlcul del TF-IDF i similitud cosinus

En aquesta segona part de la pràctica, completarem el codi del script *TFIDFViewer.py* que calcula el TF-IDF dels termes dels documents per calcular el vector de pesos i obtenir la similitud cosinus entre dos documents.
L'única dificultat que hem trobat en completar el codi ha estat en la funció que calcula la similitud cosinus perquè al principi no havíem tingut en compte l'ordre alfabètic per recórrer els vectors dels documents i el cost de l'algorisme no era linear sinó que era quadràtic. El que fem és sumar les similituds entre els documents per a obtenir la mitja de similitud entre dues carpetes.

Per veure si el programa funciona correctament el que hem fet és comprovar la similitud d'un document amb ell mateix. Com calia esperar el resultat que ens va donar va ser 1.0.
Una altra prova que vam fer consisteix en comprovar el resultat de la similitud entre els documents 3 i 4 de la carpeta docs. El resultat que ens va donar coincideix amb el de les transparències de teoria.

Per a la part d'experimentació hem seleccionat 5 directoris de la carpeta *20_newsgroups*, 5 de la carpeta *arxiv_abs* i tots els documents de la carpeta *novels*. Hem seleccionat 50 documents de cadascun d'aquests directoris per comparar-los amb 50 més de la mateixa carpeta (news, abs). Per als documents de la carpeta *novels* els hem comparat tots.
Els resultats són els següents:

Arxiu 1 - Arxiu 2	Cosine similarity
astro-ph.updates.on.arXiv.org cond-mat.updates.on.arXiv.org	0.018550240000000023
astro-ph.updates.on.arXiv.org cs.updates.on.arXiv.org	0.014070148000000003
cs.updates.on.arXiv.org hep-th.updates.on.arXiv.org	0.013952916
hep-ph.updates.on.arXiv.org physics.updates.on.arXiv.org	0.018476184000000017
physics.updates.on.arXiv.org quant-ph.updates.on.arXiv.org	0.020417356
alt.atheism sci.space	0.015541608000000031
sci.crypt sci.space	0.017709643999999945
comp.os.ms-windows.misc comp.windows.x	0.022362692000000038
comp.sys.mac.hardware comp.windows.x	0.016612759999999963
talk.politics.guns talk.religion.misc	0.02505876000000008
novels	0.02343325987144164