Popa Alexandru-Matei
342C3

Tema 3 – Inteligenta Artificiala

Pentru rezvolvarea acestei teme, am implementat solutii pentru sumarizare extractiva de stiri. Setul de date folosit este BBC News Summaries, pe care l-am impartit in cinci categorii: business, entertainment, politics, sport si tech. Am structurat fisierele folosind o clasa, care este formata dintr-un articol si sumarizarea aferenta articolului. In continuare, am impartit fiecare categorie de stiri in liste pentru antrenare formate din 75% din totalul de stiri si liste pentru testare formate din 25% din totalul de stiri. In principal am folosit biblioteca NLTK.

Pentru algoritmul ***Naive Bayes***, problema de clasificare este daca propozitia curenta va face parte din rezumat sau nu. Astfel, am calculat frecventa aparitiilor cuvintelor care apar in toate rezumatele si frecventa aparitiilor cuvintelor care apar in articole, dar nu si in rezumate. Aceste frecvente le-am folosit in functia principala de *predict*, in care mai intai elimin titlul articolului pentru o parsare mai buna a textului, apoi impart articolul in propozitii. Iterez prin fiecare propozitie, iar pentru fiecare cuvant din propozitie calculez cele doua valori, pentru cele doua clase specificate mai sus. Daca valoarea pentru clasa, in care propozitia face parte din rezumat, este mai mare decat cealalta, atunci adaug propozitia in rezumat. Aceleasi functii le aplic pentru 2-grame si 4-grame, in care, fata de 1-grame, doar concatenez cuvintele.

Pentru algoritmul ***TF-IDF***, am facut o functie care calculeaza frecventa unui cuvant intr-un document. Aceasta returneaza numarul de aparitii al cuvantului respectiv in document. Pentru algoritm am mai avut nevoie de o functie care intoarce numarul de documente dintr-o care contin cuvantul respectiv, impartit la lungimea colectiei de documente. La fel ca la Naïve Bayes, si aici iau fiecare cuvant din fiecare propozitie in parte, pe care il lematizez si adaug imbunatarile specifice din enuntul problemei. La final, am ales ca rezumatul sa fie format din primele cu cel mai bun scor - ***len(articol) / 2*** propozitii. Pentru a calcula scorul doar pentru substantive, m-am folosit de tag-ul cuvintelor din NLTK, iar pentru similaritatea cu titlul am facut - numarul de elemente din intersectia dintre titlu si propozitie impartit la numarul de cuvinte din titlul. Am ales sa inmultesc acest raport cu **1.5**, insa cred ca este o valoare prea mica, care nu influenteaza foarte mult scorul, dupa cum se va observa in graficele de mai jos. Pentru ponderarea pozitiei propozitiei in articol, am facut raportul dintre indexul propozitiei in articol si numarul total de propozitii; am ales sa inmultesc raportul cu 2.

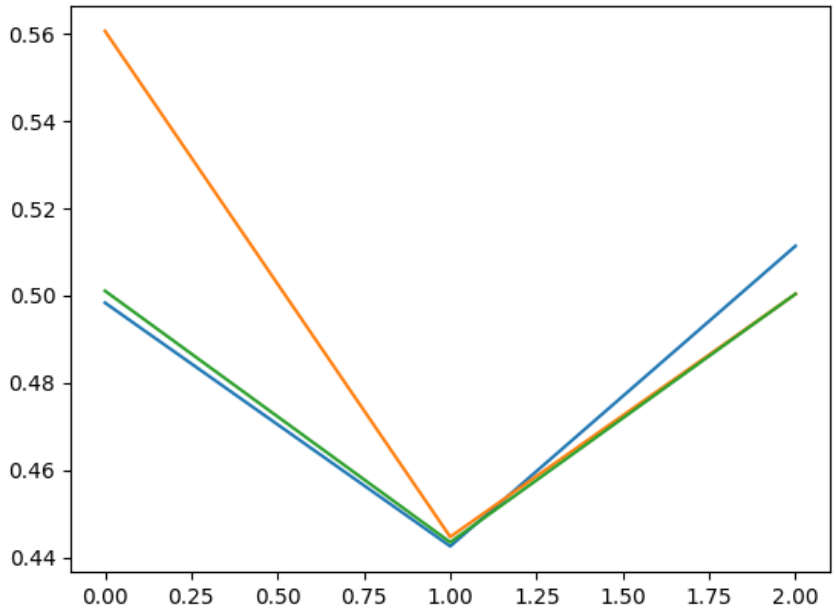
Popa Alexandru-Matei
342C3

Pentru a calcula valorile Blue si Rouge, am folosit clasa Rouge din python, care contina functia get_scores, care intoarce precizia si recall-ul pentru fiecare articol sau pentru o colectie de articole.

Naive-Bayes

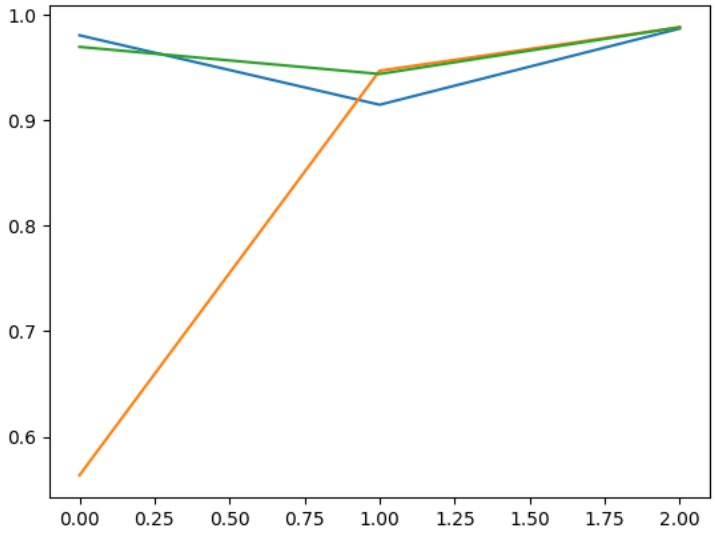
Pentru grafice am folosit ca valori – media valorile Blue si Rouge pentru articolele din categoria de testare business, cand se aplica lematizarea, eliminarea cuvintelor neinformative si pastrarea cuvintelor neinformative. Fiecare grafic este pentru monograme, bigrame si 4-grame, iar punctele care il formeaza sunt media valorilor pentru lematizare, eliminare stopwords si pastrare stopwords.

Valorile Blue pentru fiecare caz (eliminare stopwords, lematizare etc)



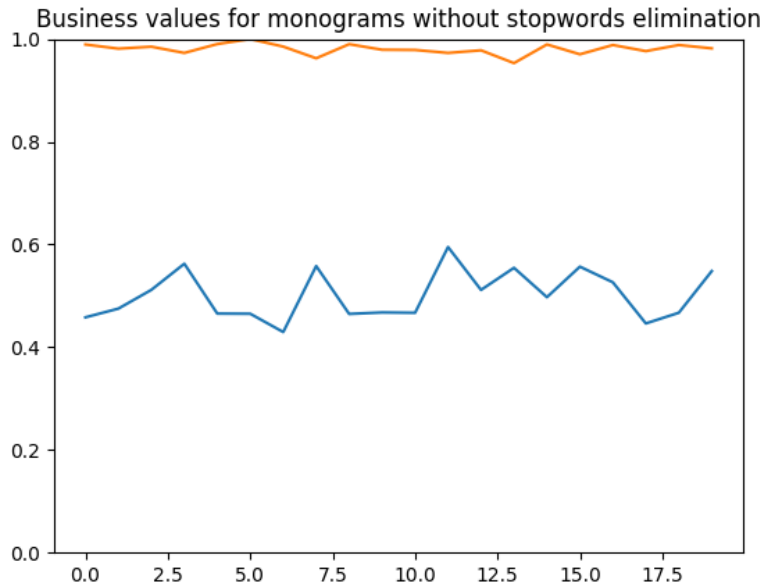
Popa Alexandru-Matei
342C3

Valorile Rouge pentru fiecare caz



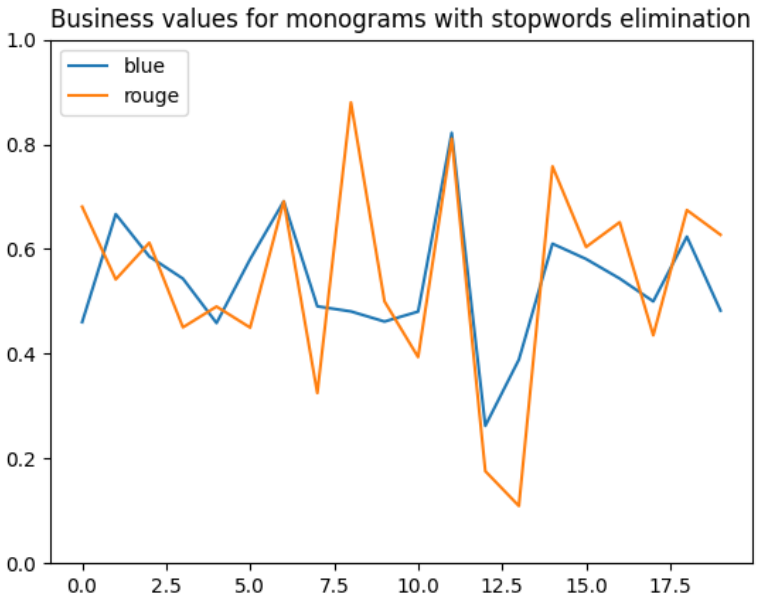
In continuare pentru grafice am folosit primele 20 de articole din categoria de testare business.

Monograme – pastrare cuvinte neinformative

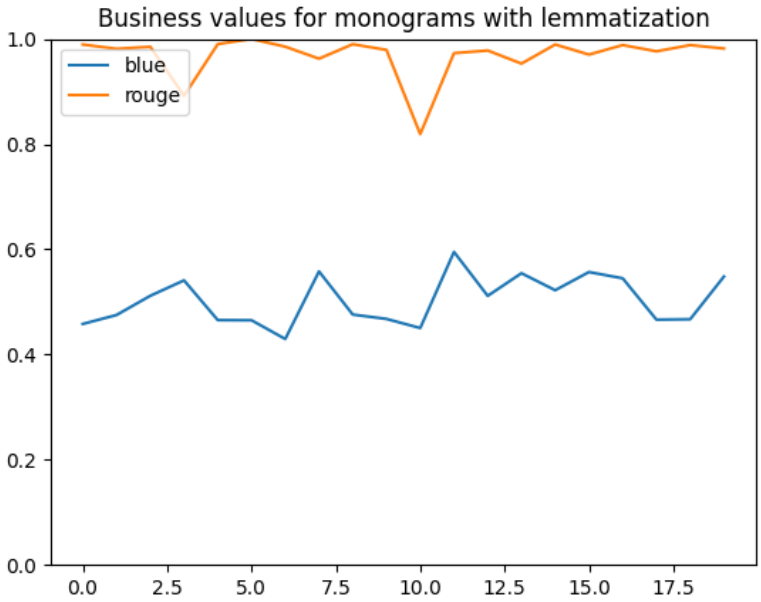


Popa Alexandru-Matei
342C3

Monograme – eliminare cuvinte neinformative

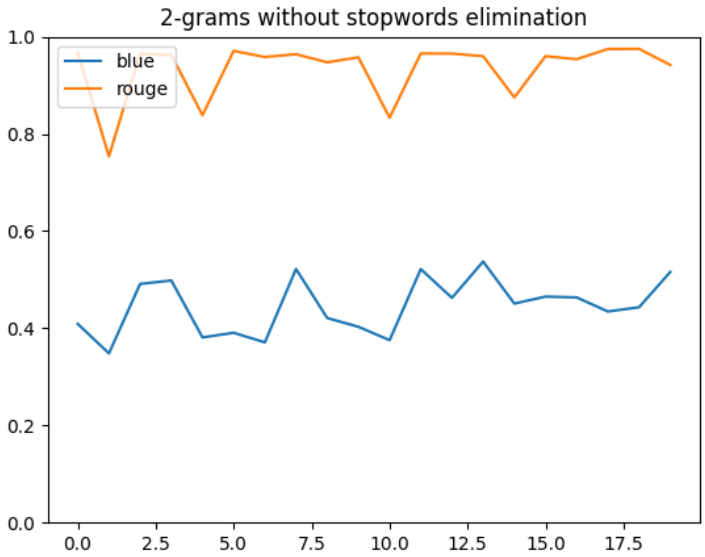


Monograme – cu lematizare

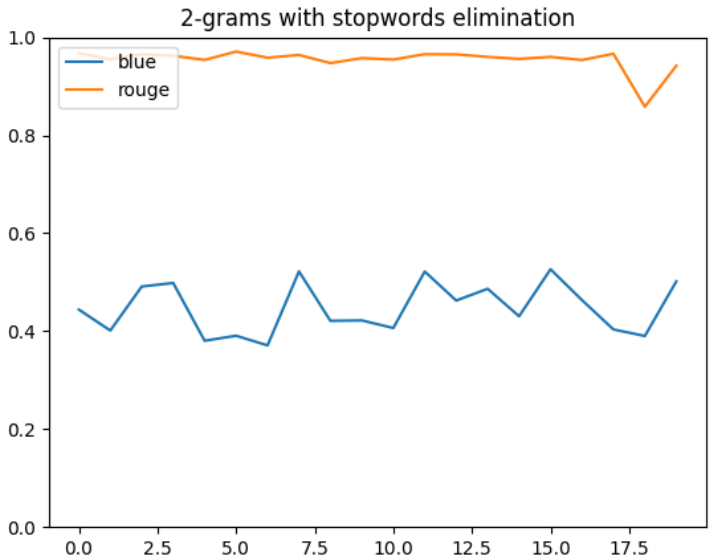


Popa Alexandru-Matei
342C3

Bigrame – pastrare cuvinte neinformative

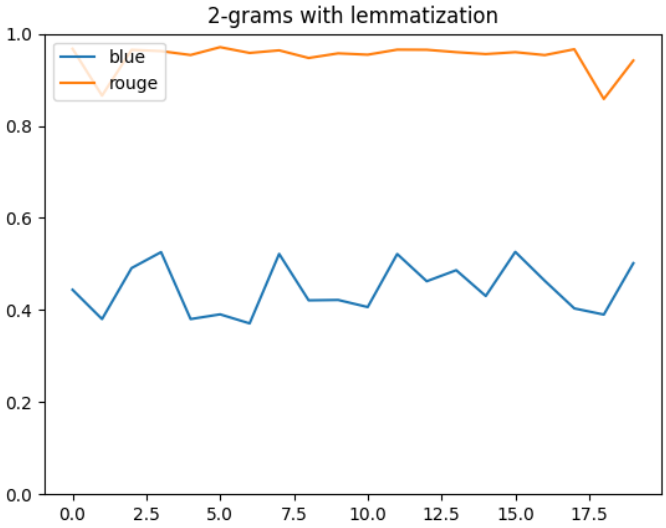


Bigrame – eliminare cuvinte neinformative

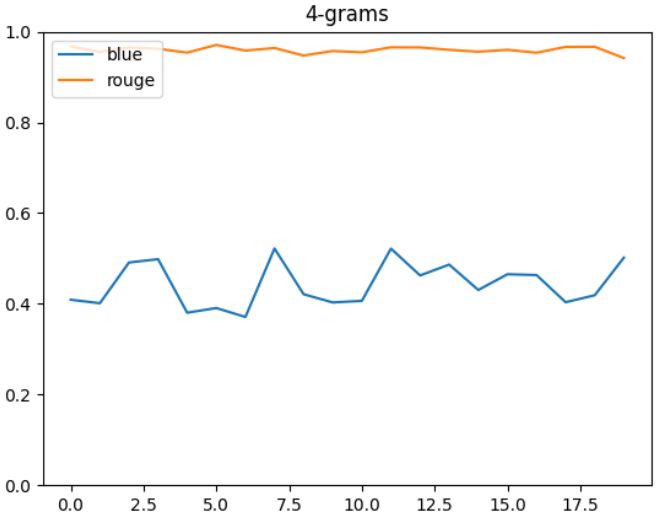


Popa Alexandru-Matei
342C3

Bigrame – cu lematizare



Pentru 4-grame, valorile erau aproape identice pentru fiecare caz – eliminare stop words, pastrare stop words, lematizare. Am ales sa fac un singur grafic.



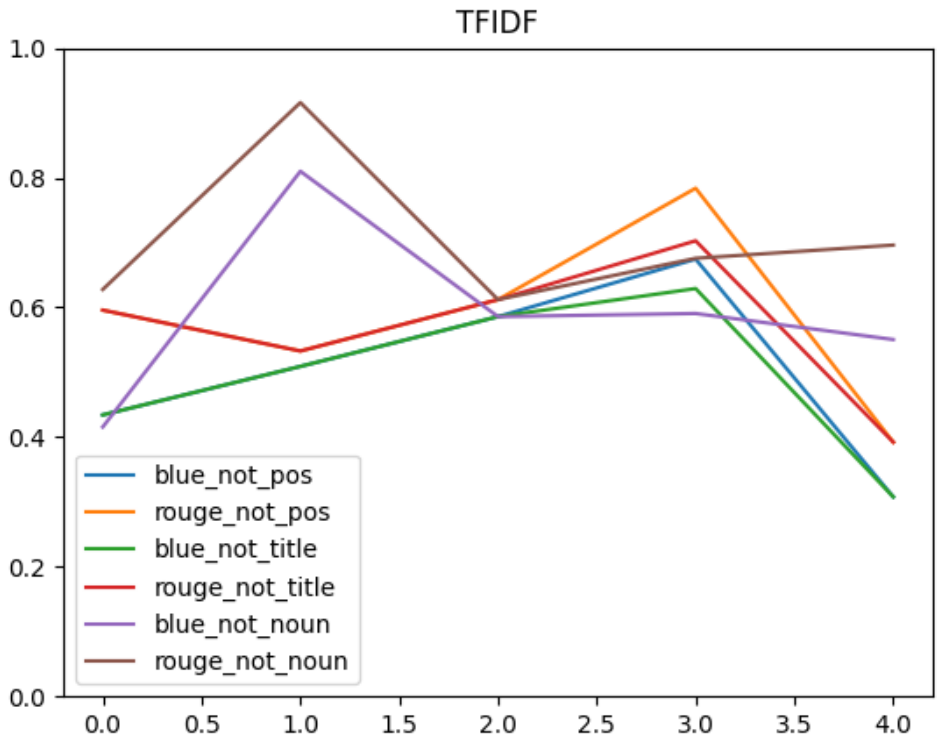
Popa Alexandru-Matei
342C3

Pentru TF-IDF am adaugat in acelasi grafic toate valorile pentru toate cazurile – pentru monograme.

Not_pos = nu se ia in considerare imbunatatirea adusa de ponderarea pozitiei propozitiei in articol

Not_title = nu se ia in calcul imbunatariea adusa de cuvintele identice cu cele din titlu

Not_noun = scorul se calculeaza pentru toate cuvintele, nu numai pentru substantive



Popa Alexandru-Matei
342C3

Concluzii:

Se observa ca scorurile bleu si rouge sunt mai mici in cazul in care este eliminata prima conditie. De aici as putea trage urmatoarea concluzie: a calcula scorul numai pentru substantive ajuta cel mai mult la precizia rezumatului. Sunt scoruri foarte asemanatoare in cazul excluderii conditiei cu titlul si cea cu pozitia propozitiei, deoarece cred ca am luat o pondere prea mica. Ele sunt mai putin semnificative decat calcularea scorului doar pentru substantive.

In cazul algoritmului Naïve-Bayes, pentru 1-grame, 2-grame si 4-grame, se observa ca precizia este mai buna cand sunt eliminate cuvintele neinformative, apoi cand se aplica lematizarea, iar, in final, daca se pastreaza cuvintele neinformative.