Tema 3 – Sumarizarea documentelor
Inteligenta Artificiala
Nicolescu Radu-Catalin, 343C4

1. Detalii de implementare

Clasa Document va fi folosita pentru a descrie un articol, avand 3 variabile: content – continutul articolului, summary – sumarul articolului, dtype – tipul articolului (sport, politics, entertainment, tech, business).

Clasa Editor este folosita pentru a prelucra continutul articolelor, aceasta ocupandu-se cu impartirea unui articol in propozitii, impartirea unui articol in cuvinte, eliminarea cuvintelor de legatura (stop words) si lematizarea cuvintelor. Aceasta are drept membru o lista de cuvinte de legatura, care este populata prin citirea fisierului stop_words dat in arhiva temei.

Clasa Loader se ocupa cu incarcarea documentelor in memorie si salvarea acestora intr-un document in format .csv, care va fi ulterior folosit in clasificare si sumarizare. Aceasta are ca variabile 5 liste de documente, cate una pentru fiecare tip de articole. Clasa poate citi toate articolele si sa le salveze local, dar si sa le salveze intr-un document csv.

Modulul runner.py se ocupa cu sumarizarea si clasificarea documentelor, folosindu-se de documentul .csv generat de Loader. Documentul .csv contine 3 coloane si un numar de linii egal cu numarul de stiri existente. Cele 3 coloane sunt content, summary si type.
Primele doua (content, summary) vor fi folosite in algoritmul Naïve Bayes de sumarizare, in timp ce tuplul (content, type) va fi folosit in algoritmul Naïve Bayes de clasificare.

Pentru clasficare, se calculeaza probabilitatile ca articolul sa apartina fiecarei dintre cele 5 clase, la final alegandu-se clasa cu probabilitatea maxima de apartenenta.

Pentru sumarizare, pentru fiecare articol, se calculeaza probabilitatile ca fiecare propozitie sa apartina sumarizarii, cat si probabilitatea medie a acestora. Propozitiile care au probabilitatile peste aceasta medie vor fi incluse in sumarizarea aferenta articolului.

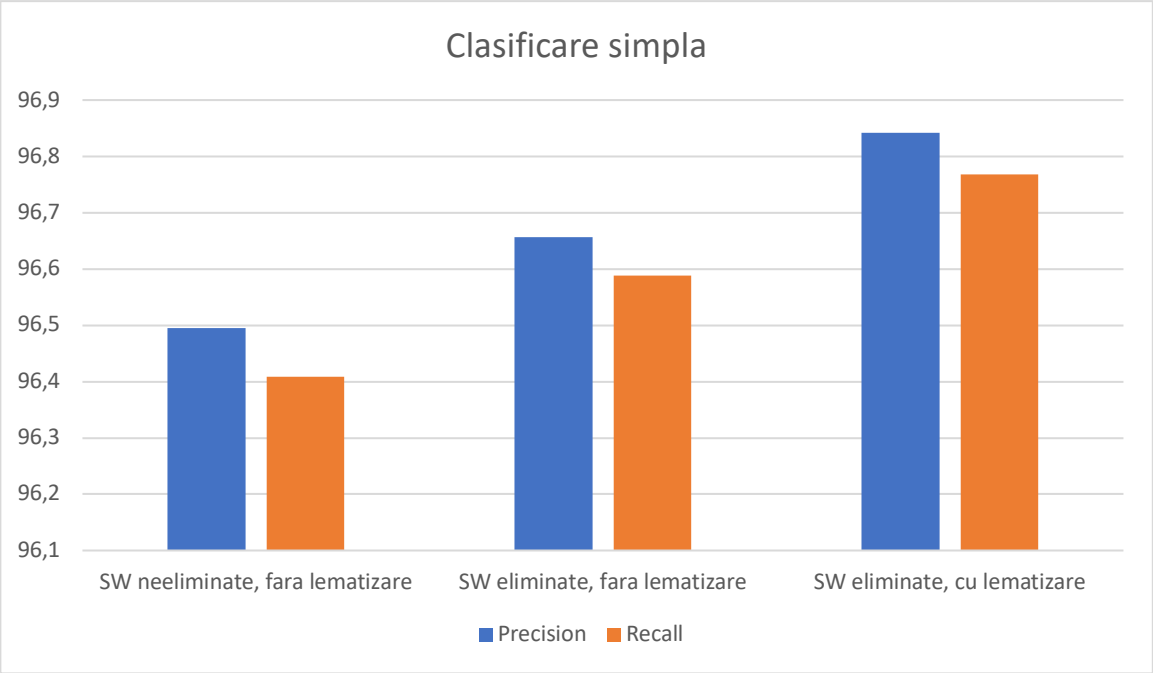
La finalul algoritmului de clasificare, se calculeaza matricea de confuzie, precizia si recall-ul algoritmului pentru 3 instante ale problemei: fara eliminare de cuvinte de legatura si fara lematizare, cu eliminare de cuvinte de legatura si fara lematizare, cu eliminare de cuvinte de legatura si cu lematizare.

La finalul algoritmului de sumarizare, se calculeaza precizia si recall-ul pentru 3 instante ale problemei: fara eliminare de cuvinte de legatura si fara lematizare, cu eliminare de cuvinte de legatura si fara lematizare, cu eliminare de cuvinte de legatura si cu lematizare, in 2 cazuri: folosind unigrame si folosind bigrame.

Pentru realizarea bonusului, am impartit setul de date in 5 parti egale, si am folosit pe rand cate o parte pentru testare, iar restul de 4 parti pentru antrenare. La final, am calculat media si deviatia standard pentru precizie si recall, pentru toate cazurile mentionate mai sus, atat pentru clasificare, cat si pentru sumarizare.

2. Rezultate experimentale

Clasificare simpla



Matricile de confuzie:

a) SW neeliminate, fara lematizare

	business	entertainment	politics	sport	tech
business	110	0	3	0	4
entertainment	1	96	2	0	3
politics	4	0	97	0	0
sport	0	0	0	134	0
tech	0	0	3	0	100

b) SW eliminate, fara lematizare

	business	entertainment	politics	sport	tech
business	110	0	3	0	4
entertainment	1	97	1	0	3
politics	4	0	97	0	0
sport	0	0	0	134	0
tech	0	0	3	0	100

c) SW eliminate, cu lematizare

	business	entertainment	politics	sport	tech
business	110	0	3	0	4
entertainment	1	96	2	0	3
politics	4	0	97	0	0
sport	0	0	0	134	0
tech	0	0	1	0	102

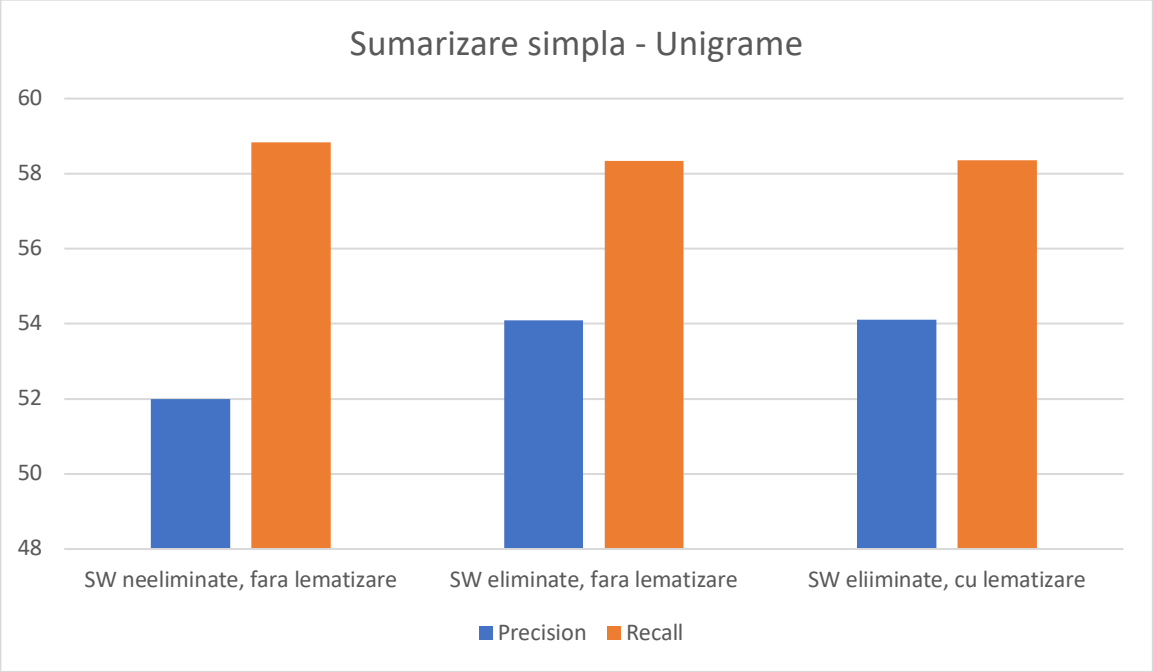
Comparand matricile de confuzie pentru cele trei cazuri, este usor de observat imbunatatirile de rezultate de la o implementare la alta: numarul de elemente de pe diagonala principala (true positives) creste, ceea ce duce la cresterea preciziei si recall-ului de la o implementare la alta.

Eliminarea cuvintelor de legatura are ca efect eliminarea din calcule a cuvintelor care nu au informatie relevanta. (nu contribuie la clasificarea catre o anume clasa)

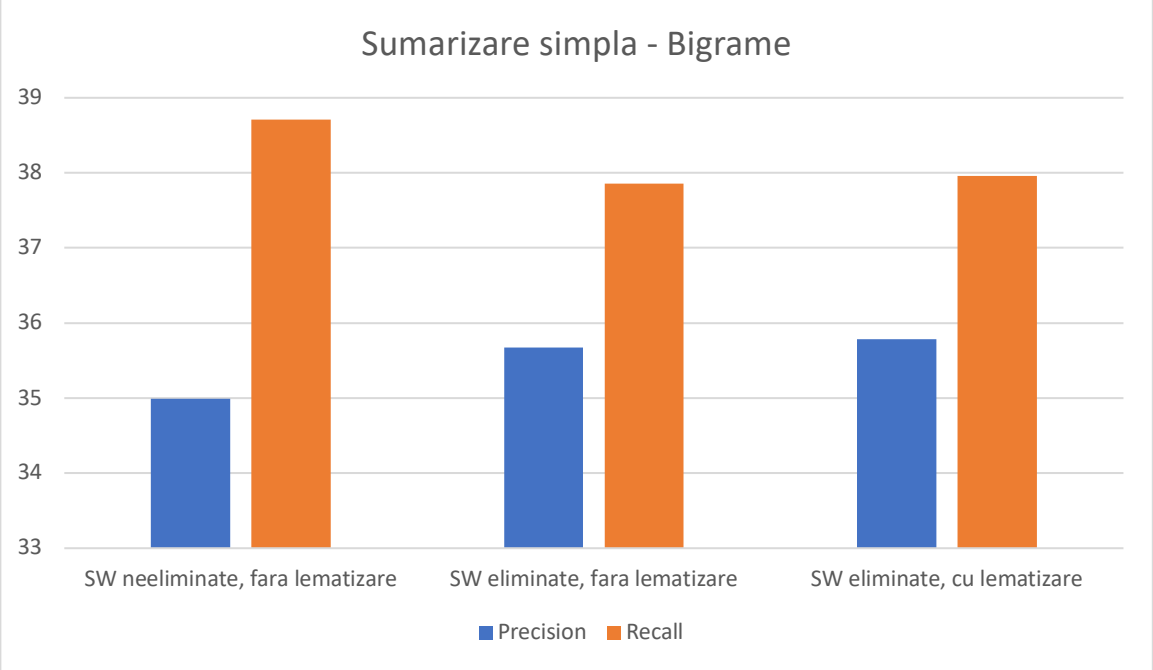
Lematizarea are ca efect cresterea numarului de cuvinte care au informatie relevanta, deoarece sunt eliminate sufixele, pluralurile. Astfel, un cuvant care apare sub mai multe forme va aparea, dupa lematizare, intr-o singura forma.

Sumarizare simpla

Unigrame



Bigrame



Performantele sumarizarii folosind unigrame sunt net superioare sumarizarii folosind bigrame, datorita faptului ca unigramele (formate dintr-un cuvant) au mult mai multe aparitii in texte decat bigramele (formate din doua cuvinte vecine), ceea ce duce la cresterea relevantei celor dintai in sensul includerii unei propozitii intr-un sumar.

Clasificare prin 5-fold cross-validation

	PRECISION		RECALL	
	Mean %	Standard Deviation %	Mean %	Standard Deviation %
SW neeliminate, fara lematizare	97,304	0,004	97,213	0,004
SW eliminate fara lematizare	97,605	0,004	97,528	0,004
SW eliminate cu lematizare	97,696	0,004	97,617	0,004

Sumarizare prin 5-fold cross-validation

UNIGRAME

	PRECISION		RECALL	
	Mean %	Standard Deviation %	Mean %	Standard Deviation %
SW neeliminate, fara lematizare	51,469	0,005	58,920	0,004
SW eliminate fara lematizare	53,121	0,005	58,412	0,002
SW eliminate cu lematizare	53,329	0,006	58,067	0,002

BIGRAME

	PRECISION		RECALL	
	Mean %	Standard Deviation %	Mean %	Standard Deviation %
SW neeliminate, fara lematizare	34,601	0,007	38,917	0,007
SW eliminate fara lematizare	35,262	0,006	38,077	0,004
SW eliminate cu lematizare	35,274	0,006	38,099	0,003

Rezultatele preciziei si recall-ului obtinute in urma 5-fold cross-validation sunt foarte apropiate de rezultatele obtinute in urma impartirii clasice a setului de date in 75% date de antrenare si 25% date de testare, fiind diferente de sub 1% intre acestea.