UNIVERSITATEA „ALEXANDRU IOAN CUZA” din IAȘI

Facultatea de Economie și Administrarea Afacerilor – Master Data Mining

**Identificarea limbajului ofensator în limba română**

**Echipă:**

Coman Alexandru Ionuț

Irina Maria-Elena

Rotaru Matian Alexandru

Ungureanu Gabriel

**IAȘI 2022**

**1. Scurtă descriere a proiectului**

Scopul proiectului este de a identifica limbajul ofensator în limba română și de a clasifica ”gravitatea” acestuia: *warning* pentru un limbaj deranjant, *offensive* pentru un limbaj deja către limita de a fi considerat obscen. Algoritmul pe care urmează să-l prezentăm pas cu pas utilizează tehnici de Machine Learning aplicate pe o serie de texte preluate din diverse canale de *social media* pentru a clasa limbajul folosit de utilizatori.

**2. Arhitectura – schema proiectului**

Diagram, schematic

Description automatically generated

Plecând de un set de texte extrase manual din diverse forumuri și platforme de socializare și de la un set de cuvinte *offensive* din limba română, realizăm un algoritm de clasificare ce identifică tipul acestui text pe baza unor parametri stabiliți în prealabil.

**Detalii la nivelul proiectului**

**1. Motivația**

În contextul curent în care o proporție semnificativă de ”discuții” au loc în online, știm că oamenii prind curaj în spatele calculatorului și sunt darnici în vorbe jignitoare în momentul în care o veste sau o părere nu corespunde cu propriile idei. Astfel, a ajuns necesară crearea de algoritmi care să identifice aceste persoane fără cenzură și să descurajeze ulterior comportamentul nepotrivit al acestora. Chiar dacă spațiul media este, teoretic, liber pentru ca oricine să își expună părerea, răutățile nu întârzie să apară niciodată.

Așadar, motivația din spatele acestui proiect o reprezintă dorința de a ne bucura cu toții de un mediu online cât mai sigur, lipsit de hărțuire și de limbaj ofensator la tot pasul.

**2. Provocări**

Printre obstacolele întâlnite în realizarea proiectului enumerăm următoarele:

* **Time Management**: nu am excelat pe partea acesta pe parcul semestrului, astfel încât task-urile s-au aglomerat toate în ultimele săptămâni de activitate didactică;
* **Resursele academice pentru informare:** au fost, pe deoparte limitate la număr și, pe de altă parte, în limba engleză. Având ca target identificarea limbajului ofensator în limba română, majoritatea resursele nu ne-au oferit informații utile decât despre modul de abordare al proiectului, nu și despre tool-urile/ pachetele care ne pot ajuta pentru un algoritm de clasificare cu o acuratețe cât mai bună;
* **Dezechilibrul la nivelul cunoștințelor tehnice**: venind din medii academice diferite și cu skill-uri care aveau o vagă tangență cu lucrul în *python* pentru procesarea datelor text, a reprezentat o adevărată provocare să realizăm o împărțire a task-urilor astfel încât acestea să se muleze pe skill-urile fiecărui membru, dar să vină și cu o notă de noutate pentru a extinde cunoștințele inițiale;
* **Construirea bazei de date**: o adevărată provocare deoarece am plecat de la ideea că avem nevoie de un set de texte deja adnotate pe categoriile pe care dorim să le clasificăm ulterior. Problema a fost că dacă am fi plecat de la un set în engleză, tradus cu un API de la Google în română, traducerile mai mult ca sigur ar fi fost aproximative, implicit de calitate redusă, iar de aici și eventualele probleme ce puteau apărea în clasificarea efectivă. Astfel, am ales să mergem pe o cale manuală – au fost extrase texte din diverse forumuri, platforme de socializare, grupuri cu interese comune etc.
* **Lucrul cu pachete cu documentație sumară**: pentru că proiectul a implicat texte în limba română, pachetele, algoritmii și documentațiile uzuale au trebuit să fie înlocuite cu structuri adaptate noii provocări.

**3. State of the art**

Deoarece nu am identificat un studiu ce să evidențieze întocmai aceeași temă, am ales să ne raportăm la o lucrare ce prezintă procesul de creare al unui algoritm ce identifică limbajul ofensator în limba engleză.

Ca punct de pornire, am ales lucrarea unor cercetători nemți – lucrare realizată în mai 2020 pentru LREC (*Language Resources and Evaluation Conference*) ce a avut loc în perioada 11-16 Mai în Marseille. Articolul a fost publicat ulterior la paginile 137-143 în *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying.*

Lucrarea pleacă de la premisa că majoritatea rețelelor de socializare utilizează un proces de moderare a conținutului public, vizibil, cu ajutorul căruia diverși moderatori verifică dacă utilizatorii oferă un limbaj ofensator sau care să încalce regulile de utilizare a platformei. În ultimii ani s-a tot adus în discuție automatizarea acestui proces sub definirea unui task de clasificare supervizat.

În lucrarea colectivă a celor 3 cercetători germani aceștia analizează și compară 4 modele diferite de clasificare a limbajului ofensator:

* Un model de învățare automată interpretabil (*naive Bayes*);
* O metodă de explicație independentă de model (*LIME*);
* O metodă de explicație bazată pe model (*LRP*);
* Un model auto-explicativ (*LSTM cu un ”attension mechanism”*)

Aplicând fiecare metodă pentru a evidenția caracterul explicativ al fiecăreia, aceștia au identificat că modelele cele mai complexe asigură o clasificare cu o acuratețe mult mai ridicată, oferind în același timp o explicare mult mai bună decât modelele simple utilizate în analiză.

**4. Descrierea modulelor și a limbajelor folosite**

Pentru construirea proiectului a fost folosit limbajul de programare *python.* Principalele pachete utilizate au fost: *pandas, numphy, sklearn* și *spacy.*

Pentru început, au fost construite două dataset-uri:

* Primul conține cuvinte cu tentă ofensatoare/ obscenă din limba română. Punctul de plecare a fost baza de date *Hatebase[[1]](#footnote-1)*, însă fiindcă aceasta nu conținea tot ”repertoriul” românesc de cuvinte ofensatoare, echipa a completat cu o serie nouă de cuvinte și expresii pe baza cunoștințelor personale și pe baza celor existente pentru fiecare literă din alfabet în română;
* Al doilea set de date conține texte preluate manual din diverse rețele de socializare, forumuri și grupuri de Facebook. Textele sunt din toate categoriile, nu conțin doar cuvinte cu intenții rele.

Atât cuvintele ofensatoare, cât și textele, au fost adnotate în prealabil. Cuvintele sunt clasificate în *warning* și *offensive,* iar textele în *not offensive, warning* și *offensive.* În partea de preprocesare, textele au fost împărțite în cuvinte *(tokens)* și au fost eliminate stopwords, numerele, cuvintele de lungime 1 inclusiv spații și puncte; de asemenea am transformat cuvintele în forma de baza *(lematizare)* rămânând astfel cu cele mai semnificative cuvinte.

Fișierul de intrare conține textele cu label-urile noastre, iar parametrii de clasificare sunt stabiliți ca interval prin intermediul primului scrip ce găsește numărul optim pentru aceștia. Spre exemplu, putem obține un interval de la 2 până la 5 cuvinte *warning* pentru ca un text să fie clasificat drept offensive.

Pentru clasificarea efectivă a textelor pe baza celor două corpusuri au fost utilizate o serie de metode diferite:

* *Decision Tree on TF-IDF matrix;*
* *Random Forest on TF-IDF matrix;*
* *SVM on TF-IDF matrix;*
* *Clasificare pe baza numărului de cuvinte offensive/ warning.*

Setul de training conține 70% din înregistrările din fișierul de intrare, iar setul de testing – restul de 30%. Fiecare metodă folosește clasificarea inițială și pe baza celor mai buni parametri identificați la începutul programului face match între numărul de cuvinte găsite din fiecare categorie și intervalul de referință calculat de program. Rezultatele sunt apropiate din punct de vedere al acurateței pentru cele 4 metode utilizate.

În ceea ce privește *output-ul*, spre deosebire de fișierul de intrare acesta deține și:

- o coloană cu textul după ce au fost eliminate stopwords;

- o coloană cu numărul de cuvinte offensive;

- o coloană cu numărul de cuvinte warning;

- o coloană cu clasificarea textului.

**5. Statistici – algoritmi utilizați**

Vom descrie pe scurt metodele utilizate în clasificare:

* **Decision Tree:** algoritmul construit pe baza AD (Arbori de Decizie) generează modele de clasificare/ estimare folosind elemente de *machine learning* și *statistică.* Ideea de bază este următoarea: algoritmul utilizează un criteriu de divizare pentru a determina cel mai predictiv factor pe care să-l amplaseze în rădăcină ca prim punct de decizie în arbore și, în continuarea, să execute o căutare a factorilor predictivi pentru a construi subarbori – toate acestea până în momentul în care nu mai sunt date de procesat. În cazul nostru, pe rădăcini se vor găsi condiții pentru numărul de cuvinte warning/ offensive pentru setul de date analizat;
* **Random Forest:** este o metodă de clasificare ce are la bază *ansambluri.* Metodele bazate pe ansambluri funcționează pe principiul conform căruia un grup de clasificatori slabi pot forma, împreună, un clasificator puternic. În cazult Random Forest, se pornește de la mai mulți arbori de decizie obținuți pe baza setului de date inițial și fiecare arbore în parte se aplică pe noua instanță ce trebuie clasificată, rezultând astfel câte o clasă pentru fiecare arbore. Rezultatul final al clasificării este dat de clasa majoritară;
* **SVM (Support Vector Machine):** este un algoritm de învățare automată supravegheat, care este frecvent utilizat pentru provocările de clasificare și regresie. Aplicațiile obișnuite ale algoritmului SVM sunt sistemul de detectare a intruziunilor, recunoașterea scrisului de mână, predicția structurii proteice, detectarea steganografiei în imagini digitale etc., însă funcționează și pe detectarea aparițiilor cuvintelor dintr-un corpus în altul.

**Project management**

**1. Alocarea membrilor echipei pentru fiecare componentă a proiectului**

Componentele proiectului sunt următoarele:

1. Corpus / Bază de date cu texte ce au fost încadrate manual în categoriile (Not Offensive / Warning / Offensive);

Listă de cuvinte (Warning / Offensive) clasificate manual;

1. Script R eliminare diacritice, cuvinte de tip „stop”, a numerelor și cuvintelor de lungime 1;

Realizare algoritm de clasificare după numărul de cuvinte Offensive/Warning;

1. Creare matrice de ponderi tf-idf pe baza textelor preprocesate folosind librăria sklearn;
2. Algoritmii Decision Tree, Random Forest și SVM (sklearn) pe baza matricei tf-idf și prezentarea rezultatelor obținute.

**2. Team leader**

Echipa noastră nu a avut un team leader. Am reușit să ne coordonăm prin vot în privința tuturor deciziilor pe care a trebuit să le luăm în demersul realizării proiectului în ceea ce privește structura acestuia, deadline-urile și împărțirea task-urilor.

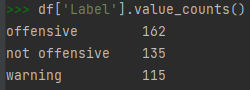
**3. Obiective prezentate detaliat**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Denumire Task | Responsabili | Start Task | Durată | End Task | Status |
| 1. Documentare | | | | | |
| Documentare bază de date/API/corpus | Maria, Alex, Marian, Gabi | 3/01/2022 | 1 | 3/01/2022 | 100% |
| Documentare listă cuvinte offensive în limba Română | Maria, Alex, Marian, Gabi | 3/01/2022 | 1 | 3/01/2022 | 100% |
| Documentare algoritmi de clasificare | Maria, Alex, Marian, Gabi | 3/01/2022 | 1 | 3/01/2022 | 100% |
| 2. Planificare și execuție | | | | | |
| Replanificare a task-urilor | Maria, Alex, Marian, Gabi | 5/01/2022 | 1 | 5/01/2022 | 100% |
| Realizarea listei de cuvinte offensive/warning | Maria, Alex, Marian, Gabi | 6/01/2022 | 2 | 7/01/2022 | 100% |
| Realizarea corpusului | Maria, Alex, Marian, Gabi | 7/01/2022 | 3 | 10/01/2022 | 100% |
| Adnotarea listei de cuvinte offensive/warning | Alex, Maria | 11/01/2022 | 2 | 12/01/2022 | 100% |
| Adnotarea corpusului | Alex, Maria, Marian | 11/01/2022 | 2 | 12/01/2022 | 100% |
| Întrunire pentru replanificare task-uri | Maria, Alex, Marian, Gabi | 13/01/2022 | 1 | 13/01/2022 | 100% |
| Modificare corpus și listă badWords (adăugare înregistrări) | Alex, Marian | 14/01/2022 | 1 | 14/01/2022 | 100% |
| Documentare și execuție script eliminare diacritice | Alex | 11/01/2022 | 1 | 11/01/2022 | 100% |
| Documentare și execuție script stop words | Gabi | 15/01/2022 | 1 | 15/01/2022 | 100% |
| Stabilire parametrii inițiali de clasificare | Maria, Alex, Marian, Gabi | 11/01/2022 | 1 | 11/01/2022 | 100% |
| Realizare algoritm de clasificare | Maria, Alex, Marian, Gabi | 12/01/2022 | 4 | 15/01/2022 | 100% |
| Documentare și execuție matrice de ponderi tf-idf pe baza textelor preprocesate | Maria, Alex, Marian, Gabi | 15/01/2022 | 2 | 16/01/2022 | 100% |
| Documentare algoritmi pentru analiză statistică | Marian | 15/01/2022 | 2 | 16/01/2022 | 100% |
| Implementare algoritmi Random Forest și Decision Tree | Maria, Alex, Marian, Gabi | 15/01/2022 | 2 | 16/01/2022 | 100% |
| Implementare algoritm SVM (sklearn) | Marian | 16/01/2022 | 1 | 16/01/2022 | 100% |
| 3. Finalizare | | | | | |
| Agregare module realizate pentru întocmirea arhivei finale | Maria, Alex, Marian, Gabi | 17/01/2022 | 1 | 17/01/2022 | 100% |
| Creare repository pe GitHub și upload fișiere | Alex | 18/01/2022 | 1 | 18/01/2022 | 100% |
| Realizare documentație conform indicații | Maria, Alex | 5/01/2022 | 16 | 19/01/2022 | 100% |
| Realizare prezentare proiect PPT | Maria, Alex | 18/01/2022 | 2 | 19/01/2022 | 100% |
| Revizuire și turn-in | Maria, Alex, Marian, Gabi |  |  |  |  |

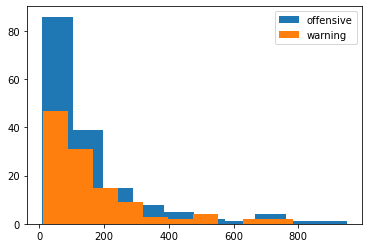
**4. Date calitative și cantitative pentru analiza proiectului**

În procesul de realizare a corpusului ne-am propus să strângem cel puțin 300 de înregistrări/texte. Am reușit să strângem 412 printr-un efort de echipă, de pe diferite site-uri web cu review-uri, secțiuni de comentarii de pe rețelele de social media și forumuri. Ulterior, textele au fost clasificate/adnotate tot manual.

Pentru realizarea și clasificarea listei de cuvinte „badWords” am apelat inițial la un corpus online (<https://hatebase.org/>) care nu s-a dovedit a fi de foarte mare ajutor într-un final, din cauza numărului restrâns de înregistrări puse la dispoziție. Așadar, tot pe cale muncitorească am luat pe rând fiecare literă din alfabet și împreună am adunat o listă de 509 cuvinte ce se încadrează într-una din categoriile Warning sau Offensive pentru cazul nostru.



După rularea algoritmului de clasificare observăm că avem 162 de înregistrări clasificate ca offensive, 135 ca not offensive și 115 ca warning. Putem spune că distribuția este relativ uniformă, așa cum ne așteptam.



În urma adnotării și rulării algoritmului de clasificare, observăm conform imaginii de mai sus că în medie, textele labeled ca offensive tind să aibă un număr mai mare de cuvinte în corpusul nostru.

La partea de acuratețe a clasificării, rezultatele sunt următoarele:

1. **44.1%** - clasificare pe baza nr de cuvinte Offensive/Warning
2. **43.5%** clasificare cu Decision Tree
3. **43.5%** clasificare cu Random Forest
4. **42.7%** clasificare cu SVM

1. <https://hatebase.org/search_results/language_id=ron%7Cpage=1> [↑](#footnote-ref-1)