Componenta B – Dinica Mircea

Analiza problemei:

Studiul 1:

<https://www.researchgate.net/profile/Ryan-Boyd-8/publication/358725479_The_Development_and_Psychometric_Properties_of_LIWC-22/links/6210f62c4be28e145ca1e60b/The-Development-and-Psychometric-Properties-of-LIWC-22.pdf>

Prezinta toolul folosit pentru preprocesarea textului. Categorizeaza cuvinte in diverse emotii si parti gramaticale. Ne ofera 114 feature-uri.

Stiudiul 2:

<https://www.proquest.com/docview/2407636676?sourcetype=Scholarly%20Journals>

Compara api-urile pentru translatare. Concluzia este ca google translate este bun, dar este platit, iar Yandex este cel mai bun neplatit. Din pacate, de la realizarea studiului politicile Yandex s-au schimbat asa ca am ales sa folosesc o librarie din python numita googletrans care foloseste google translate.

Studiul 3:

<https://www.researchgate.net/profile/Md-Siraj-Doulah/publication/368691213_Performance_Evaluation_of_Machine_Learning_Algorithm_in_Various_Datasets/links/63f5ea130d98a97717ad3a4e/Performance-Evaluation-of-Machine-Learning-Algorithm-in-Various-Datasets.pdf>

Compara mai multi algoritmi de clasificare pe mai multe dataseturi. Concluzia este ca random forest este cel mai bun. Problema este ca niciunul din dataseturi nu are atat de multe feature-uri cate avem noi.

Solutia propusa:

Dataset: <https://www.kaggle.com/datasets/infamouscoder/depression-reddit-cleaned/data>

Schema:

1. Translatare dataset din engleza in romana (se foloseste libraria googletrans mentionata)
2. Preprocesare(pentru fiecare dataset separate):
   1. Tokenizare bert preantrenata nevoita pentru tool, de pe hugging face
   2. Aplicare tool LIWC, se foloseste libraria liwc din python, care profita de dictionarul raw
   3. salvare datelor preprocesate
3. Antrenare folosind Random forest din sklearn (pentru fiecare dataset separate)
4. Evaluare vom prioritiza recall-ul,pe langa precizie,acuratete,f1 score, doarece fals negativele sunt importante