Racism Analysis

Natural Language Processing

Realizat de:

- Buhai Darius
- Olaru Elena
- Bouruc Petru Liviu
- Badescu Madalina

Alegerea datasetului

Pentru alegerea datasetului am optat pentru datasetul tweets_hate_speech_detection oferit de Hugging Face. Acest dataset contine 31,962 de tweet-uri colectate folosind API-ul de la Tweeter și clasificate în tweet-uri rasiste și non-rasiste.

Fiecare intrare din dataset este formată din 3 câmpuri:

- id-ul tweet-ului
- label-ul tweet-ului (0 = neutru, 1 = rasist)
- textul tweet-ului

Exemplu de intrare:

- *49,0,feeling blue #illustration*
- *57,1,@user lets fight against #love #peace*

Preprocesarea datelor

	id	tweet
0	0	@switchfoot http://twitpic.com/2y1zl - Awww, t...
1	0	is upset that he can't update his Facebook by ...
2	0	@Kenichan I dived many times for the ball. Man...
3	0	my whole body feels itchy and like its on fire
4	0	@nationwideclass no, it's not behaving at all....
5	0	@Kwesidei not the whole crew
6	0	Need a hug
7	0	@LOLTrish hey long time no see! Yes.. Rains a...
8	0	@Tatiana_K nope they didn't have it
9	0	@twittera que me muera ?

Aspecte ale preprocesării:

- Eliminarea userului
- Eliminarea link-urilor
- Eliminarea punctuațiilor, numerelor și a caracterelor speciale
- Eliminarea cuvintelor scurte
- Modificarea literelor mari în litere mici
- Eliminarea caracterelor repetitive
- Modificarea literelor mari în litere mici
- Eliminarea simbolului # din hashtag-uri
- Eliminarea spațiilor multiple
- Stemming

	id	tweet
0	0	aww that bummer shoulda david carr third
1	0	upset that update facebook texting might resul...
2	0	dived many times ball managed save rest bounds
3	0	whole body feels itchy like fire
4	0	behaving here because over there
5	0	whole crew
6	0	need
7	0	long time rains only fine thanks
8	0	nope they didn have
9	0	muera

Extragerea Feature-urilor

```
def createVocab(reviews):
    vocab = wordFrequency(reviews, min_occurences=18)
    vocab = removePunctuationAndStopwords(vocab)
    return vocab
```

Funcțiile utilizate:

- removePunctuationAndStopwords
- wordFrequency
- createVocab
- createVectorize
- pad
- vectorizeSentences

```
def createVectorize(vocab, reviews, total_features):
    word_indices = dict((c, i + 2) for i, c in enumerate(vocab))
    indices_word = dict((i + 2, c) for i, c in enumerate(vocab))
    indices_word[0] = 'UNK'
    word_indices['UNK'] = 0

    reviews_vectorized = vectorizeSentences(reviews, word_indices)
    reviews_vectorized = pad(reviews_vectorized, max_length=total_features)
    return reviews_vectorized
```

```
def pad(samples, max_length):
    return torch.tensor([
        sample[:max_length] + [1] * max(0, max_length - len(sample))
        for sample in samples
    ])
```

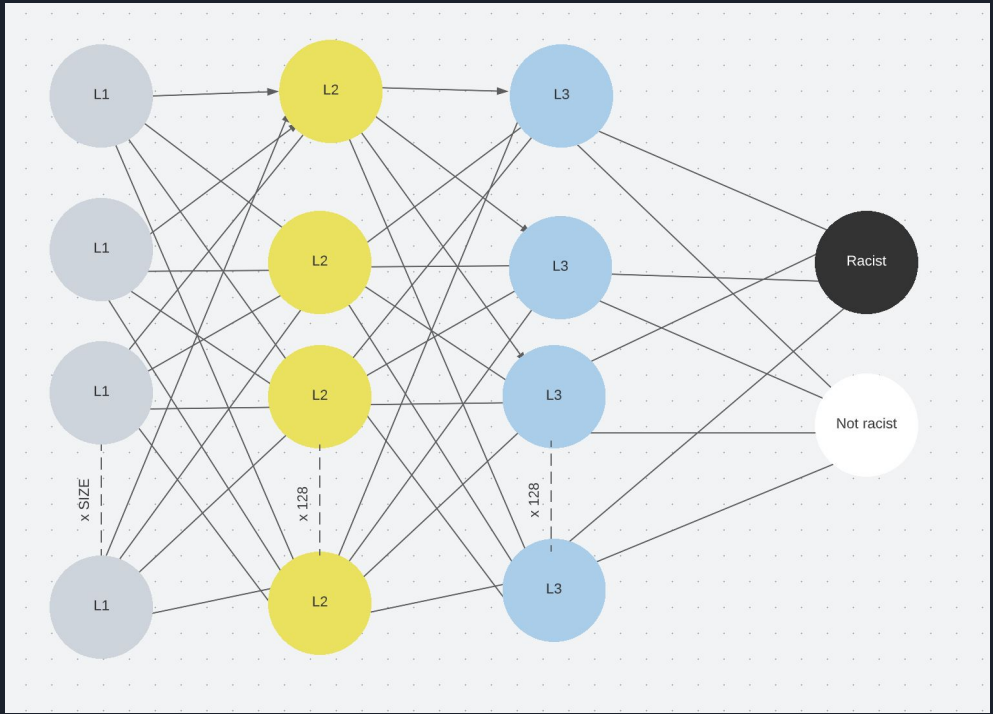
```
def vectorizeSentences(data, char_indices, one_hot=False):
    vectorized = []
    for sentences in data:
        sentences_of_indices = [char_indices[w] if w in char_indices.keys() else char_indices['UNK'] for w in sentences]
        if one_hot:
            sentences_of_indices = np.eye(len(char_indices))[sentences_of_indices]

        vectorized.append(sentences_of_indices)

    return vectorized
```

Clasificare

- Pentru clasificare am folosit un model CNN cu 2 layere convolutionale de SIZE x 128 si 128 x 128 si un average pooling 1D si un layer liniar 128 x 2 pentru output.



Articole Asociate

- „**Whose opinions matter? Perspective-aware models to identify opinions of hate speech victims in abusive language detection**” - Sohail Akhtar, Valerio Basile și Viviana Patti
- „**A survey of Race, Racism, and Anti-Racism in NLP**” - Anjalie Field, Su Lin Blodgett, Zeerak Waseem și Yulia Tsvetkov
- „**Racism Detection by Analyzing Differential Opinions Through Sentiment Analysis of Tweets Using Stacked Ensemble GCR-NN Model**” - Ernesto Lee, Furqan Rustam, Patrick Bernard Washington, Fatima El Barakaz, Wajdi Aljedaani și Imran Ashraf