Rule-Based Classification
Offensive/Not Offesive Intagram Comments
Meta 1 Project

Pedro Silva, Ramyad Raadi
uc2023235452@student.uc.pt, uc2023205634@student.uc.pt

University of Coimbra, Faculty of Science and Technologies
Bachelor of Artificial Intelligence and Data Science, PLN Course

October 2025

1 Introduction

This study presents the development of a rule-based Instagram comments classification system. The project includes data analysis, extraction of linguistic knowledge, definition of classification rules, and evaluation of system performance. The main goal is to create the most effective model possible for identifying and categorizing offensive/not offensive comments based on linguistic patterns.

2 Materials

2.1 Data Description

The dataset used for this study is publicly available on HuggingFace[1] and contains comments extracted from Brazilian politicians posts on Instagram. The data were manually annotated by three experts, it is already balanced and there are 7000 comments with the following characteristics:

[1] **Dataset (corpus):** https://huggingface.co/datasets/franciellevargas/HateBR

Column	Description
id	Comment Id
comentario	The comment itself
anotator1	The first annotator
anotator2	The second annotator
anotator3	The third Annotator
label_final	classification label: offensive (1)/not offensive (0)
links_post	The link where the comment was retrieved
account_post	Personal account of the linked post

Table 1: Dataset Column Descriptions

2.2 Tools and Packages

The project was implemented using the following tools and packages:

- **Scikit-Learn, Matplotlib, Seaborn:** Utilized for the construction of the rule-based classification system and its evaluation.
- **Pandas, re, String, NLTK, SpaCy:** for data manipulation, preprocessing and lexical analysis.

3 Exploratory Data Analysis

In this section, we analyzed the dataset and extracted relevant linguistic knowledge. The analysis includes inter-annotator agreement, lexical patterns, and sentiment characteristics of the comments.

3.1 Annotators Concordance

To assess the consistency of manual annotations, the agreement between annotators was calculated using Krippendorff's Alpha. The obtained value of $\alpha = 0.771$ indicates a medium-strong level of agreement, which is reliable for linguistic and classification analysis.

3.2 Lexical Analysis

A lexical analysis was performed on the corpus after removing punctuation, while retaining stop words and emojis. The processed dataset contained 106 015 tokens and 12 093 unique types.

An examination of the most frequent tokens was conducted under three conditions: (1) maintaining stop words and emojis, (2) analyzing stop words within the top 100 tokens, and (3) removing stop words but keeping emojis.

The results showed that emojis appear frequently among the top tokens, which is valuable for sentiment and offensive speech analysis. Removing stop words provided clearer insights into meaningful lexical patterns, allowing the classification to focus more directly on relevant linguistic features.

3.3 Sentiment Analysis

Sentiment analysis was performed using a Portuguese lexicon [2] containing words with one or two polarities per word, where -1 represents negative, 0 neutral, and 1 positive sentiment. Additionally, an annotated emoji dataset from Twitter comments [3] was used to determine the predominant sentiment of each emoji.

A preprocessing pipeline, the same used for lexical analysis and, more ahead, classification rules, was applied. Tokens were filtered to remove stop words, empty spaces, non-representative characters (e.g., the Zero-Width Joiner \u200d), and punctuation, while emojis were separated for individual sentiment assessment.

For each comment, the sentiment score was calculated as the sum of token sentiments divided by the total number of tokens. Sentiment was tested with and without lemmatization, and in some cases, lemmatization slightly altered the sentiment values. The resulting scores ranged from -1 to 1, reflecting the overall sentiment of the comment. Overall, this approach provided a sentiment estimation that correlated well with offensive/not offensive classification, supporting (what we thought) the effectiveness for this task.

4 Methods

The classification system was developed using a rule-based pipeline. The pipeline applies the preprocessing steps already discussed, followed by the application of some rules to classify comments as offensive or not.

4.1 Baseline

For the baseline model, the dataset was preprocessed and split into training and test sets (80/20). The baseline classification relied on two rules:

- **Rule 1:** Comments with strong negative sentiment (score = -1) were classified as offensive.
- **Rule 2:** Comments containing offensive words, retrieved from a Portuguese offensive words list [4] were classified as offensive.

4.2 PoS as Classification Rule

This approach extends the baseline by incorporating a Part-of-Speech (PoS) heuristic:

[2] **PT Lexicon:** `https://b2find.eudat.eu/dataset/b6bd16c2-a8ab-598f-be41-1e7aeecd60d3`

[3] **Emojis Dataset:** `https://www.kaggle.com/datasets/thomasseleck/emoji-sentiment-data`

[4] **Offensive words list (Adapted by Pedro Silva):** `https://pt.scribd.com/document/522716988/palavras-ofensivas`

- **Rule 3:** If a comment contains two or more negative adjectives or verbs, it is classified as offensive.

4.3 Cosine Similarity and Sentiment Proximation

An additional rule was introduced to leverage semantic similarity:

- **Rule 4:** Comments are compared to a set of negative seed comments (randomly sampled up to 50 extreme negative examples from the training set) using cosine similarity. If the similarity exceeds a predefined threshold, the comment is classified as offensive.

5 Results and Discussion

A series of experiments were conducted to evaluate and improve the performance of the rule-based classification system, following the rules defined in the previous section.

5.1 Baseline Results

The baseline model, which applied Rule 1 (strong negative sentiment) and Rule 2 (offensive words), achieved results that exceeded initial expectations. While F1-scores around 30% were anticipated, the model obtained F1-scores of 0.78 for the non-offensive class and 0.72 for the offensive class, with an overall accuracy of 76%. To confirm these results, multiple train-test splits were tested with different random seeds, and performance remained consistent. The model produced 271 false positives out of 1,400 comments, which was expected to be greater than the 80 false negatives, given that only explicit negative sentiment or offensive terms triggered the offensive label.

5.2 Incorporating PoS-Based Rule

Adding Rule 3 (PoS heuristics for negative adjectives and verbs) slightly improved the model's performance. The F1-scores increased to 0.80 for non-offensive and 0.76 for offensive comments, with an overall accuracy of 78%. The number of false positives decreased to 230 and the number of false negatives was similar, indicating that this rule effectively captured additional offensive content while reducing misclassifications. These results validated the relevance of including linguistic structure in rule-based classification.

5.3 Cosine Similarity Rule Evaluation

The fourth rule introduced cosine similarity to negative seed comments as a semantic approximation measure. A threshold parameter was tested to control classification sensitivity: lower thresholds reduced false negatives but increased

false positives, while higher thresholds had the opposite effect. Despite fine-tuning, the model's overall performance remained nearly identical to that of the three-rule version.

Further analysis revealed that comments not captured by the first three rules had low semantic similarity to negative seeds, making this rule ineffective for additional classification. Alternative methods for seed selection were also tested but produced similar outcomes. Therefore, Rule 4 did not contribute to performance improvement.

5.4 Overall Discussion

The best performance was obtained using the first three rules, with an accuracy of 78% and balanced F1-scores for both classes. The confusion matrix and the classification report confirmed consistent and reliable classification behavior, as we can verify in the figures 1 and 2.

All experiments were also tested with and without lemmatization in the pre-processing pipeline. The results showed that lemmatization did not improve the system performance. On the contrary, it slightly decreased all evaluation metrics and increased both false positives and false negatives. Therefore, lemmatization was excluded from the final pipeline configuration.

In conclusion, the rule-based approach proved highly effective, especially considering its interpretability and simplicity. Future work could focus on expanding the rule set to reduce false positives and maybe exploring hybrid approaches that combine linguistic rules with machine learning models to further enhance performance.

6 Acknowledgments

We would like to mention that AI [ChatGPT] was used to:

1. Correct some syntax errors and make the report more cohesive and coherent;
2. Provide insights on how to prevent data leakage between test/train sets;
3. Building complex regex;
4. Building small code components such as:
   (a) Capturing emoji patterns and spacing between emojis;
   (b) Application of cosine similarity in this context with rule classification;

7 Figures

	precision	recall	f1-score	support
NotOffensive	0.73	0.88	0.80	689
Offensive	0.86	0.68	0.76	711
accuracy			0.78	1400
macro avg	0.79	0.78	0.78	1400
weighted avg	0.79	0.78	0.78	1400

Figure 1: Classification report of the best experiment

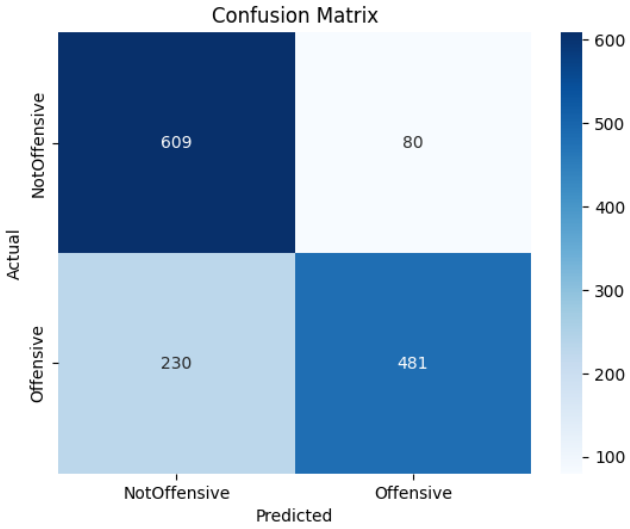

Figure 2: Confusion matrix of the best experiment

8 References

1. Hugo Oliveira, Isabel Carvalho and Patrícia Ferreira's slides and documents

2. Splitting "word1.word2": https://stackoverflow.com/questions/265960/best-way-to-strip-punctuation-from-a-string

3. Emojis Token Extraction:

   - https://www.unicode.org/Public/emoji/1.0//emoji-data.txt
   - https://gist.github.com/slowkow/7a7f61f495e3dbb7e3d767f97bd7304b
   - https://www.kaggle.com/code/stpeteishii/emoji-with-u200d

4. Sentiment analysis with Portuguese lexicon and emojis:

   - https://b2find.eudat.eu/dataset/b6bd16c2-a8ab-598f-be41-1e7aeecd60d3
   - https://b2share.eudat.eu/records/93ab120efdaa4662baec6adee8e7585f
   - https://doi.org/10.23728/B2SHARE.93AB120EFDAA4662BAEC6ADEE8E7585F
   - https://www.kaggle.com/datasets/thomasseleck/emoji-sentiment-data
   - Yoo, Byungkyu & Rayz, Julia. (2021). Understanding Emojis for Sentiment Analysis. *The International FLAIRS Conference Proceedings, 34*. https://doi.org/10.32473/flairs.v34i1.128562
   - https://www.geeksforgeeks.org/python/reading-writing-text-files-python/
   - https://stackoverflow.com/questions/517923/what-is-the-best-way-to-remove-accents-normalize-in-a-python-unicode-string

5. Portuguese offensive words:

   - https://pt.scribd.com/document/522716988/palavras-ofensivas + Pedro Silva Adaptation