**REPORT OF SENTIMENT ANALYSIS**

Students:

FERREIRA, Jesse.

TALBI, Samir.

More details on github: <https://github.com/jessefilho/sentimentanalysis>

**INTRODUTION**

This report has the purpose of put in practice the knowledge achieved during the discipline of Natural Processing Languages, diving in the yield of sentimental analysis through the text classification.

Sentimental Analysis is important to detect customers feedback, political orientation, epidemic flow and a lot of other subjects related and express via pure text. Text is the data format most available on internet and mining it give a bunch of options, however, mining data with this kind of format raise several challenges such as languages, lexical morphology, symbols and characters, etc. To make possible the sentimental analysis it necessary apply several techniques such as Tokenization, Morphological analysis, Morphological disambiguation, and so on.

For sake of simplicity this report will present 3 techniques: Tokenization by simply trimming words of a given corpus, tokenization and concatenation of Part of speech form the Morphological disambiguation and ngram. Using as weighting factor the frequency of terms to the analysis of positive or negative sentences.

**TASK 1**

**Term frequencies:**

The computation of frequency of terms is an important approach that can increase the weight over a sentence, such frequency can indicate that a sentence is important or not. This study focusing on the simple computation of terms to the first part of Task 1.1, where we study the tokenization by simply trimming words. Thus, In the Task 1.2, we did terms weighting based on its POS (Part-Of-Speech), with the purpose of highlight the adjectives (JJ) and verbs (VB). Such kind of words can be helpful to distinguish whether a sentence is positive or negatives.

**Code:**

**A screenshot of a cell phone

Description automatically generated**

* 1. **- Simply Trim (without Part-Of-Speech):**

We used the basic trim offered as support script.

* 1. **- Classifiers used and results with term trimmed:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Train Precision (%)** | **Test Precision (%)** |
| LibLinear | 86.375 | 87.5 |
| SVM SPegasos | 88.375 | 88 |
| RandomFlorest | 81.6875 | 70.5 |

**2.1 – Part-Of-Speech Trimmed:**

We used the basic trim offered as support script.

**2.2 - Classifiers used and results with term Part-Of-Speech Trimmed:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Train Precision (%)** | **Test Precision (%)** |
| LibLinear | 80.0625 | 76.5 |
| SVM SPegasos | 85.125 | 75 |
| RandomFlorest | 86.5625 | 83.5 |

**TASK 2**

**NGRAM:**

**Codes:**

1.1 – **Trimming with ngram:**