TWITTER SENTIMENT ANALYSIS COMPARISON

Supervisor: Lect. Dr. Gabriel Iuhasz

Student: Adrian-Ioan Tuns

SENTIMENT ANALYSIS

- What is Sentiment Analysis?
- Twitter Particular Case
- Common Approaches:
  - Machine learning approach
  - Lexicon-based approach

DATASETS USED

- Both Datasets Were Taken From Kaggle
- First dataset:
  - size of 8.559 KB, with 100.000 rows
  - 3 columns: “ItemID”, “Sentiment”, “SentimentText
- Second dataset:
  - size of 233.307 KB, with 1.600.000 rows
  - 6 columns: “target”, “id”, “date”, “flag”, “user”, “text”

NAÏVE BAYES CLASSIFIER

- Supervised Learning Algorithm
- Highly Suited for Text Classification
- Based on the Bayes Theorem

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- Where:
  - P(A|B) is the probability of hypothesis A on the observed event B
  - P(B|A) is the probability of the evidence given that the probability of a hypothesis is true
  - P(A) is the probability of hypothesis before observing the evidence
  - P(B) is the probability of evidence

PREPROCESS PHASE

- Every tweet is converted to lower case and the Twitter usernames, the punctuations, numbers, special characters, extra spaces, single letters and URLs are removed
- The repetitions of more than 2 letters were reduced to 2 letters and some of the most common emojis were analyzed and converted to positive and negative emojis
- The contractions of the words are replaced (e.g., can't => can not)
- The text is normalized with lemmatization, which transforms the inflected forms of a word to a root form (e.g., playing => play, played => play)
- The stop words, which represent very common words (e.g., me, was, to), are removed
- The newly obtained processed tweets are saved in a new column, called "processed_tweet"

CLASSIFICATION PHASE

- Separate (tokenize) the words of each processed tweet
- Create a count vector of the words appearing in the text, by taking into consideration the Unigrams (single words) only
- Compute the Inverse Document Frequency (IDF), which is a measure of the importance of a term, that gives a larger weight for the terms which are less common in the corpus
- Split the dataset in training and test sets (with 80% of the datasets as training, and 20% as test)
- Train the multinomial Naïve Bayes classifier
- Predict the sentiment of the test subsets
- Measure the accuracy of the model by comparing the predictions with the actual values and by using specific evaluators

IMPLEMENTATION AND EXPERIMENTS

- Preprocess step
  - Common step, implemented in Python
- Sequential implementation
  - In Python, with scikit-learn (sklearn) library
  - Run locally on a Window machine, from PyCharm IDE
- Distributed implementation
  - In Python, with Apache Spark, through PySpark interface
  - Run locally on a Window machine, from PyCharm IDE
  - Run on Google Cloud, using a Dataproc cluster on Computer Engine

RESULTS

- Sequential implementation, locally:
  - Small dataset execution time: 9.780968427658081 seconds, accuracy: 73.735693036857 76%
  - Large dataset execution time: 15.19605541229248 seconds, accuracy: 76.998942184248 28%
- Distributed implementation, locally:
  - Small dataset execution time: 8.445001363754272 seconds, accuracy: 70.964522728406 76%
  - Large dataset execution time: 31.253251552581787 seconds, accuracy: 75.749772891019 02%
- Distributed implementation, on Google Cloud Dataproc:
  - Small dataset execution time: 11.35384798049 seconds, accuracy: 70.99830457764037%
  - Large dataset execution time: 48.59863781929016 seconds, accuracy: 75.70725957876325%

CONCLUSION

- Comparison Between Two Implementations of a System to Classify the Sentiment of a Tweet Was Made
- Premise Was Not Met
- Possible Factors:
  - Dataset size too small
  - Inefficient implementation of the algorithm
  - The preprocessing step done separately greatly affected the measured time

THANK YOU FOR YOU ATTENTION!

BIBLIOGRAPHY

- Dataproc | Google Cloud. (2023, January 28). Retrieved from https://cloud.google.com/dataproc
- PyCharm. (2023, January 28). Retrieved from https://www.jetbrains.com/pycharm
- scikit-learn: machine learning in Python — scikit-learn 1.1.2 documentation. (2023, January 28). Retrieved from https://scikit-learn.org/stable
- About Twitter | Our company and priorities. (2023, January 28). Retrieved from https://about.twitter.com/en
- Walaa Medhat, Ahmed Hassan, Hoda Korashy, Sentiment analysis algorithms and applications: A survey, Ain Shams Engineering Journal