Tarea 3 – ISIS 4221

Natural Language Processing

Coding rules: Use jupyter notebooks and be sure that the notebook is executed and contain the results before submitting. All classes, methods, functions and free-code MUST contains docstrings with a detail explanation. Build a notebook for each point.

Report: Together with the notebooks, you must submit a written report (please use pdf format) with the answers to the questions and a short summary of the implementation.

Submission: Assignments are submitted via Bloque Neon. Do not email us your assignments. Please upload all files and documents. You can work in pairs or individually.

Datasets

- **20N: 20Newsgroups (http://qwone.com/~jason/20Newsgroups/)**
- **Multi-Domain Sentiment Dataset (http://www.cs.jhu.edu/~mdredze/datasets/sentiment/)**

PLEASE READ DATASET DESCRIPTIONS

You can download all datasets from:

https://www.dropbox.com/sh/pzukl8aztgecvio/AAAqHBPfpH8lqQLOqbIGVfOHa?dl=0

[50p] Naive Bayes (NB), Logistic Regression (LR)

You can use existing implementations of NB and LR, as well as evaluation metrics. I recommend https://scikit-learn.org/.

I. For the 20N dataset compare two classifiers NB and LR to identify the 20 different newsgroups.
   - Create your own processing pipeline for the task and justify it.
   - Divide the dataset into training (60%), validation (10%) and test (30%).
   - Train NB and LR using the following vector representations:
     - tf (counts) representation (sklearn: CountVectorizer).
     - tfidf representation (sklearn: TfidfVectorizer).

II. Investigate cross-validation technique.
   - Explain what the strategy consists of and what it is used for.
   - Compare the results of NB and LR using 10-fold cross validation:
     - Use for cross validation: training+validation sets.
     - Do a search for LR hyperparameters (i.e. learning rate).
   - Report precision, recall, and F1 with the macro and micro average results.

III. Evaluate models using the test set:
    - Report precision, recall, and F1 with the macro and micro average results.
    - What is the best model?

[50p] Sentiment Analysis

I. Use "Multi-Domain Sentiment Dataset" to build a sentiment classifier (positive/negative) per each category ("Books", "DVD", "Electronics", "Kitchen").
    - Use negative.review+positive.review as training+validation dataset.
    - Use unlabeleded.review as testing datasets.
    - Report the results using NB as LR as classification algorithms over the test set, using as evaluation metrics precision, recall, F1, and accuracy. Use the following features representation strategies:
        - tf (counts) representation.
        - tfidf representation.
        - Features only extracted from lexicons[1]. Please document which features you built with enough detail.
    - Compare and analyze results in terms of:
        - NB vs LR
        - Features representation.
        - ***Categories ("Books", "DVD", "Electronics", "Kitchen"). Which category is more difficult to predict sentiment?, why?***
        - According to LR parameters what are the most important features per category?

II. Repeat the process but instead of building a classifier per category, build a single classifier for all categories.
    - Merge all categories and build a consolidate training+dev, and testing dataset.
    - Report the results using NB as LR as classification algorithms over the test set, using as evaluation precision, recall, and F1. Use the following features representation strategies:
        - tf (counts) representation.
        - tfidf representation.
        - Features only extracted from lexicons.
    - Compare results in terms of:
        - NB vs LR
        - Features representation.
        - **One vs multiple classifiers. Is it worth building a classifier for each category? justify your answer.**
    - According to LR parameters what are the most important features? compare with those obtained in I.

[1] In the dropbox link you can find some English lexicons. You are free to use any lexicon (there are many) and use them to create other features.