CE807 – Assignment 2 – Sentiment analysis

School of Computer Science and Electronic Engineering - University of Essex

Assignment Due at 11:59:59am on 23-Apr-2019

Electronic Submission URL:

https://www1.essex.ac.uk/e-learning/tools/faser2/

Please also see your student handbook for rules regarding the late submission of assignments

On Plagiarism

The work you submit must be your own. Any material you use, whether it is from textbooks, classmates, the web or any other source must be acknowledged in your work.

MOTIVATION: Sentiment analysis forms a core part of *Text Analytics*, and a nice illustration of the many different facets and industrial applications in this research field is the programme of the *Sentiment Analysis Symposium* 2018 (see http://2018.sentimentsymposium.com/). The topics range from sentiment in market research through crowdsourcing product design via social media, to artificial emotion in customer engagement. Sentiment analysis is not going away soon and there is huge demand for people like you, who understand the fundamental principles and can apply them appropriately in practical settings.

OBJECTIVE: The objective of this assignment is to get some practical experience in designing and running your own sentiment classifier. The dataset will be the Rotten Tomatoes movie review dataset, a commonly used resource. Each `document' in this dataset represents a single sentence from the reviews and the task is to train a classifier that will be able to classify previously unseen sentences.

SUBMISSION, ASSESSMENT AND RULES

- This assignment counts towards 50% of the overall mark for CE807.
- The assignment is to be done individually or in pairs. If you work in pairs, you each need to submit the same report (please include information about which two reports should be treated as a pair). Both members of a pair will get the same mark unless there is reason to do otherwise.
- Be sure to put your name and registration number as a comment at the top of all code and other files.
- The assignment must be submitted in a single zipped archive containing the following subfolders:

CE807/Assignment2/	All files
CE807/Assignment2/Task1	The report produced for Task 1.
CE807/Assignment2/Task2	The code written to extract features in Task 2 (with comments describing the features you used) and the CSV file produced for the *Kaggle* competition named as follows: ce807_assignment2_givenname_surname.csv The filename should be all lower case, no use of spaces, replacing *givenname* and *surname* by your own name, e.g. ce807_assignment2_alba_garcia.csv
CE807/Assignment2/Task3	The report produced for Task 3.

Note: you are free to use any software you like for this assignment. Your software should run on your laptop or in one of the CSEE labs.

Sentiment Analysis on Movie Reviews

Classifying the sentiment of sentence is one of the standard tasks in text analytics. The objective of the assignment is to develop a tool to classify the sentiment of individual sentences from a standard collection benchmark for sentiment analysis, the *Rotten Tomatoes* dataset.

Data description

The Rotten Tomatoes movie review dataset is a corpus of movie reviews used for sentiment analysis, originally collected by Pang and Lee (Pang & Lee, 2005) and it is available from

https://www.kaggle.com/c/sentiment-analysis-on-movie-reviews

The dataset (more details on the Kaggle site) is comprised of tab-separated files with phrases from the Rotten Tomatoes dataset. The train/test split has been preserved for the purposes of benchmarking, but the sentences have been shuffled from their original order. Each sentence has been parsed into many phrases by the Stanford parser. Each phrase has a *PhraseId*. Each sentence has a *SentenceId*.

- train.tsv contains the phrases and their associated sentiment labels.
- test.tsv contains just phrases. You must assign a sentiment label to each phrase.

The sentiment labels are:

0 - negative
1 - somewhat negative
2 - neutral
3 - somewhat positive
4 - positive

Getting started

You will first have to register with Kaggle (https://www.kaggle.com/) and join the competition "Sentiment Analysis on Movie Reviews" (https://www.kaggle.com/c/sentiment-analysis-on-movie-reviews). In order to participate in the competition, just follow the instructions.

Tasks

Your tasks will be as follows:

- In Task 1, you will **review the literature** to produce a concise critical discussion of the state of the art in Sentiment Analysis focusing primarily on approaches evaluated using movie reviews, specifically the *Rotten Tomatoes* datasets. We expect this to be no longer than two pages plus references.
- In Task 2, you will **devise and train your own classifier using a suitable set of features** extracted from the given dataset. You can use any tools you like for extracting the features, e.g. any of the tools you have come across in the labs such as NLTK, ScikitLearn etc. You can also use any classification tool you like. To evaluate your solution, you will have to submit your classification of the test set to the competition. Submissions are evaluated automatically by the system using classification accuracy (the percent of labels that are predicted correctly) for every parsed input line.
- In Task 3, you will **write a report** explaining what you did in Task 2 together with some motivation (why you did it) and reflection (which alternatives did you consider). You will be asked to compare and contrast your results with what you found in Task 1.

_____ TASKS _____

TASK 1: Sentiment analysis literature review

Whenever you develop a new classifier or other text analytics software, you have to show that your system outperforms the state of the art. The goal of this part of the assignment is to explore the landscape of approaches that have been applied to sentiment analysis for movie reviews, in particular the given dataset. This review should highlight what classification algorithms and features have led to the best classification accuracy. You should also identify simple approaches that can easily be implemented and which serve as **simple baseline systems**. An example of a baseline system could be one that simply uses tokens as features. We would expect that your own classifier will outperform a simple baseline system but will likely not manage to obtain results of state-of-the-art approaches.

You can start by looking at the two papers referenced at the competition site.

TASK 2: Devising and training your own classifier

This task will only be marked if you have completed Task 1.

For this task, you will develop your own classifier using the features of your choice and the classification algorithm of your choice. The pre-processing steps and classification tools you have come across in the labs should get you started. Doing this will involve:

- Identifying the features you want to extract;
- Developing code to extract these features from text (and weigh them if you want to use more than simply binary features);
- Train a classifier that uses these features;
- Evaluate the performance of the classifier using your test set by submitting it to the competition.

TASK 3: Report

This task will only be marked if you have completed Tasks 1 and 2.

Finally, you will write a report documenting what you did in Task 2 and comparing and contrasting your approach with what you discussed in Task 1. You should explain why you decided on the algorithm and features you used and how this compares to the state of the art. You should discuss the performance of your approach and reflect on what you have learned.

Bibliography

Pang, B., & Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *Association for Computational Linguistics.*

MARKING BREAKDOWN (out of 100%)

Task 1. Sentiment analysis literature review (20%)

- Appropriate coverage and contextualisation: up to 10%
- Critical discussion: up to 10%

Task 2. Design and implementation of a sentiment classifier (50%) - Task 1 required

- Design suitable algorithm and features: up to 10%
- Training the classifier using a suitable framework: up to 10%
- High-quality code including comments up to 10%
- Achieving state-of-the-art performance: up to 10%
- Material submitted as requested: up to 10%

Task 3. Report (30%) – Task 2 required

- Discussion of work carried out: up to 10%
- Comparison with the state of the art: up to 10%
- Lessons learned: up to 10%