

Natural Language Processing: Assignment 3

Due: Sunday 18 June 2017 at 23:59 | Submission via PointCarré or e-mail

Summary

In this final assignment, you will implement your own sentiment analyser to predict the polarity (positive/negative) of the sentiment that is expressed by individual consumers in their online reviews of products they bought. Reviews consist of a short evaluative text and a product score (1 to 5) that are assumed to express the same underlying sentiment. Your sentiment analyser will learn to predict the reviewer's score from the written text, so that afterwards, the system can also detect the sentiment in reviews without explicit scores. You have considerable liberty for making your own choices in the different parts of the assignment. First, you can choose which data to train and test your system on by selecting reviews from 1 specific product category out of 24 different ones. Next, you can make personal and creative choices in the design of your system's architecture: You can choose which text pre-processing components to include, which machine learning algorithm you use for training a probabilistic sentiment analyser, and which textual features this algorithm should learn from. Depending on the algorithm, your sentiment analyser will assign a sentiment class or predict a sentiment score for unseen reviews. You will evaluate the results of your sentiment analyser on a held-out test set, using standard evaluation measures like precision and recall, and macro/micro averaged F1.

Although there are minimum requirements for each part of the assignment (see details below), the many "degrees of freedom" also give you the flexibility to choose how much time you want spend on the assignment. To reflect this in your overall score for the assignments, you can assign a higher or lower weight to Assignment 3 relative to the previous two assignments. This way, you can compensate if you think one of your previous assignments didn't go so well, or you might simply get so enthusiastic about Assignment 3 that you spend more time. The 3 assignments together can earn you 10 points of the 20 course points. By default, all assignments are weighted equally (3.33 points each). **If you want to change the weight of Assignment 3, you should state at the beginning of your report how many points you like assignment 3 to count for.** You can allocate a **minimum of 2 and a maximum of 6 points** (only use integers). The remaining points will be equally distributed over Assignment 1 and 2.

As background material to this assignment, you can consult the slides of Class 10 on PointCarré and read Chapters 6 and 18 of Jurafsky and Martin, (3rd Edition, draft) *Speech and Language Processing*. (<https://web.stanford.edu/~jurafsky/slp3/>)

Part 1: Implementing a probabilistic sentiment analyser

Dataset

For his research, Dr. [Julian McAuley](http://jmcauley.ucsd.edu/data/amazon/) at the University of California (San Diego) has collected product reviews from Amazon, covering 24 different product categories. He has publicly released samples of the datasets on his website <http://jmcauley.ucsd.edu/data/amazon/>. The datasets come in the format of JSON-dumps.

- Choose a product category that you like to work on and download the corresponding dataset.
- If you think the dataset is too large to be processed on your computer, you can take a sample (min. 10,000 reviews)

Pre-processing

- **Scores** (field "overall"): Depending on whether you choose a regression or classification approach for your sentiment analyser (see next step), you can either keep the original 5-point scale scores or discretize them into 3 sentiment categories (positive/neutral/negative), for which you can choose interval boundaries yourself, but motivate these in your report.
- **Text** (field "reviewText") Preprocess the text data using python libraries of your choice (NLTK, [spaCy](#)...). You can choose the amount of pre-processing you want to do (tokenize, POS-tag, dependency parser) but make sure that you at least handle the problem of **negation** (see section 6.4 in Jurafsky and Martin 3rd Ed.)

- Split your dataset in a training set and test set, and, if needed, a development set. (Optionally you can also make splits for 10-fold cross validation)

Training your sentiment analyser

- Choose a supervised machine learning algorithm you would like to use for training your sentiment analyser (motivate your choice in your report). You can use the Naïve Bayes classifier we discussed in class or any other type of regression and/or classification algorithm you think is suitable. You don't have to implement the algorithm yourself: use any publicly available python library you like.
- Use the training and development splits from your dataset to train and fine-tune at least 2 different models to predict the review scores and/or sentiment categories from text.
 - At least one model should use n-grams as features (you can choose the order of n-grams yourself and optionally fine-tune what works best using the dev split of your dataset)
 - At least one model should use features derived from a sentiment lexicon. You can choose any of the publicly available sentiment lexicons discussed in class and J&M Ch.18 (e.g. [SentiWordNet](#)).
 - (Depending on your pre-processing, you can optionally also experiment with features that make use of POS-tags or dependency relations)

Evaluation of your sentiment analyser

- Use the test split from your dataset to evaluate each of your models with the appropriate evaluation measures (see Jurafsky & Manning 3rd Ed. section 6.6, or any other evaluation measure you think is appropriate)
- (Optionally you can evaluate using 10-fold cross validation (see J&M Ch.6.8))
- Compare the performance of your models
- Pick out 10 misclassifications (negative classified as positive and vice-versa) for further manual error analysis. Discuss in your report why you think your sentiment analyser misclassified these reviews. Think of the classifications issues discussed in the slides (slide 31 and following).

Part 2: Report

- State how I should weight your result on this assignment (min. 2 points, max. 6 points)
- Clearly state which dataset you have chosen!
- Discuss and motivate your design choices.
- Discuss your evaluation, compare your models, report on the error analysis and formulate some (wild) ideas for further improvement

Handing in your assignment:

Submit your report **by June 18, 23:59** via PointCarré or via e-mail (kris.heylen@ai.vub.ac.be) as a **zip archive** that includes the following 7 files:

1. **report.pdf** : your report as a pdf-file
2. **Your dataset**: Your (sample of) reviews as JSON-dumps with the splits into **training, dev and test set**. (if the files are too big for upload to PointCarré, you can use index files with unique IDs per review that refer to the original dataset on Julian McAuley's website. In that case, include a mapping function that takes the index-file to extract relevant reviews from the original dataset.)
3. **sentanal.py**: code as 1 python script, easily runnable, with a main function that takes as input a file from your dataset and that outputs tuples <linenr, observed score/category, predicted score/category> written to a tab-separated output file.
4. **testout.txt**: The output file from sentanal.py for your test data set

If you have any questions about the assignment, please post them on the forum on PointCarré.

Good luck!

Kris Heylen