Document Similarity Benchmark

Subtitle

Jan-Gabriel Mylius

Heidelberg University
Institute of Computer Science
jmylius@stud.uni-heidelberg.de

January 28, 2020

Outline

Outline

Motivation

- Create a Framework for comparing similarity algorithms.
- Make it easily expandable.
- Provide commonly used algorithms.

Outline

The Architecture

The benchmark suite consists of two essential elements:

- the algorithms
- the datasets

For both of these we want to have a base class that allows easy instantiation of customized versions.

Algorithms

There is a template class for implementing custom algorithms. The following functions need to be implemented:

- train(self create_vec(self, in_line), in_dataset, in_score)
- create_vec(self, in_line)
- compare(self, a, b)

Preimplemented Algorithms

The benchmark suit offers a number of already implemented algorithms:

- bag of words
- bag of words with lemmatization
- word2vec
- bert

Datasets

The dataset class:

- Dataclasses holding test-, training-datasets and -scores.
- Methods for loading SICK- and STS-datasets.
- Methods for training and running the algorithms on the set.

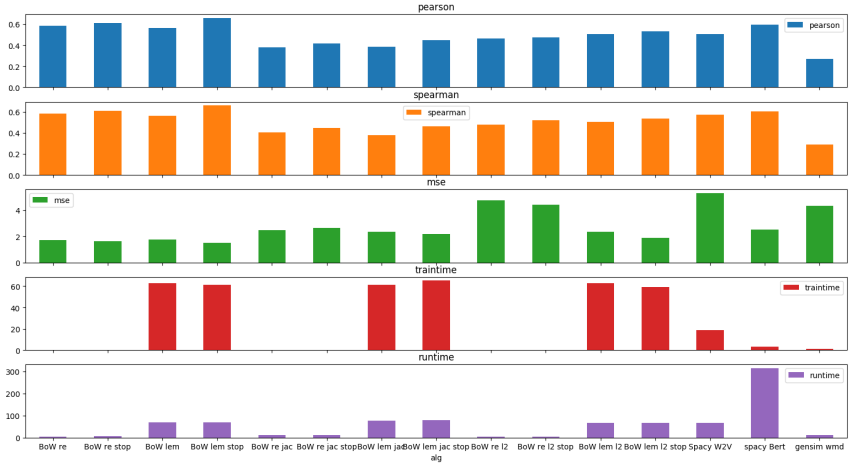
The benchmark

Via commandline the algorithms to be run can be selected.
These are then run on both datasets. The following metrics are saved to a json file:

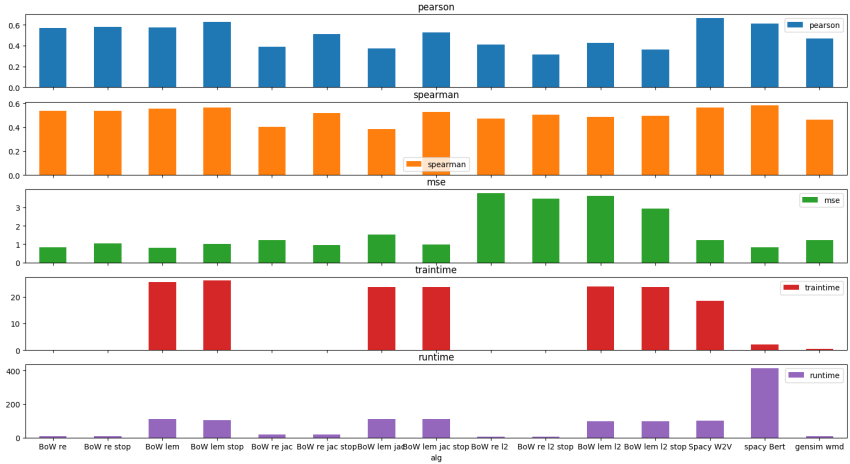
- Pearson correlation
- Spearman correlation
- Mean Squared Error
- Training Time
- Runtime

Outline

Overview of all results



Overview of all results



Discussion of Results

Lessons Learned