

 explosion / sense2vec

🔥 Use spaCy to go beyond vanilla word2vec

#spacy #nlp #natural-language-processing #word2vec #python #sense2vec

 131 commits

 2 branches

 0 releases

 5 contributors

 MIT

Branch: master ▾

New pull request










Create new file

Upload files

Find file

Clone or download ▾

 ines committed on GitHub Update .travis.yml Latest commit e3e871d on 1 Feb

 bin	pin numpy to 1.8	2 years ago
 include	add stdint.h fallback (vs 2008)	2 years ago
 sense2vec	Update about.py	11 months ago
 .gitignore	Initial commit	2 years ago
 .travis.yml	Update .travis.yml	11 months ago
 LICENSE	Update LICENSE	a year ago
 MANIFEST.in	cleanup	2 years ago
 README.rst	Update README.rst	11 months ago
 buildbot.json	ship numpy 1.7 headers	2 years ago
 requirements-all.txt	ship numpy 1.7 headers	2 years ago
 requirements.txt	Remove BLIS dependency again, due to problems. This version may be sl...	a year ago
 setup.py	Fix spaCy requirement in setup.py	a year ago

sense2vec: Use spaCy to go beyond vanilla word2vec

Read about sense2vec in our [blog post](#). You can try an online demo of the technology [here](#) and use the open-source [REST server](#).

build passing pypi v0.6.0

Overview

There are three relevant files in this repository:

bin/merge_text.py

This script pre-processes text using spaCy, so that the sense2vec model can be trained using Gensim.

bin/train_word2vec.py

This script reads a directory of text files, and then trains a word2vec model using Gensim. The script includes its own vocabulary counting code, because Gensim's vocabulary count is a bit slow for our large, sparse vocabulary.

sense2vec/vectors.pyx

To serve the similarity queries, we wrote a small vector-store class in Cython. This made it easier to add an efficient cache in front of the service. It also less memory than Gensim's Word2Vec class, as it doesn't hold the keys as Python unicode strings.

Similarity queries could be faster, if we had made all vectors contiguous in memory, instead of holding them as an array of pointers. However, we wanted to allow a `.borrow()` method, so that vectors can be added to the store by reference, without copying the data.

Installation

Until there is a PyPI release you can install sense2vec by:

1. cloning the repository
2. run `pip install -r requirements.txt`
3. `pip install -e .`
4. install the latest model via `sputnik --name sense2vec --repository-url http://index.spacy.io install reddit_vectors`

You might also be tempted to simply run `pip install -e git+git://github.com/spacy-io/sense2vec.git#egg=sense2vec` instead of steps 1-3, but it expects [Cython](#) to be present.

Usage

```
import sense2vec
model = sense2vec.load()
freq, query_vector = model["natural_language_processing|NOUN"]
model.most_similar(query_vector, n=3)
```

```
(['natural_language_processing|NOUN', 'machine_learning|NOUN', 'computer_vision|NOUN'], <MemoryView of 'nda
```



For additional performance experimental support for BLAS can be enabled by setting the USE_BLAS environment variable before installing (e.g. `USE_BLAS=1 pip install ...`). This requires an up-to-date BLAS/OpenBlas/Atlas installation.

Support

- CPython 2.6, 2.7, 3.3, 3.4, 3.5 (only 64 bit)
- OSX
- Linux
- Windows