WordEmbeddingLoader Documentation

Release 0.2.1

Yuta Koreeda

Nov 05, 2017

Modules

Loaders and savers for different implentations of word embedding. The motivation of this project is that it is cumbersome to write loaders for different pretrained word embedding files. This project provides a simple interface for loading pretrained word embedding files in different formats.

```
from word_embedding_loader import WordEmbedding

# it will automatically determine format from content
wv = WordEmbedding.load('path/to/embedding.bin')

# This project provides minimum interface for word embedding
print wv.vectors[wv.vocab['is']]

# Modify and save word embedding file with arbitrary format
wv.save('path/to/save.txt', 'word2vec', binary=False)
```

This project currently supports following formats:

- GloVe, Global Vectors for Word Representation, by Jeffrey Pennington, Richard Socher, Christopher D. Manning from Stanford NLP group.
- **word2vec, by Mikolov.**
  - text (create with `-binary 0` option (the default))
  - binary (create with `-binary 1` option)
- gensim 's `models.word2vec` module (coming)
- original HDFS format: a performance centric option for loading and saving word embedding (coming)

Sometimes, you want combine an external program with word embedding file of your own choice. This project also provides a simple executable to convert a word embedding format to another.

```
# it will automatically determine the format from the content
word-embedding-loader convert -t glove test/word_embedding_loader/word2vec.bin test.
↪bin

# Get help for command/subcommand
word-embedding-loader --help
word-embedding-loader convert --help
```

CHAPTER 1

Issues with encoding

This project does decode vocab. It is up to users to determine and decode bytes.

```
decoded_vocab = {k.decode('latin-1'): v for k, v in wv.vocab.iteritems()}
```

CHAPTER 2

Development

This project us Cython to build some modules, so you need Cython for development.

`bash pip install -r requirements.txt `

If environment variable `DEVELOP_WE` is set, it will try to rebuild `.pyx` modules.

`bash DEVELOP_WE=1 python setup.py test `

CHAPTER 3

CHANGELOG

3.1 v0.2.1

- bugfix:

** Loading binary word2vec fails with python3 (Issue #6)

3.2 v0.2

- Supports for python 3.4+
- `WordEmbedding.vocab` stores words as bytes instead of unicode.

** This allows more consistent loading/saving without needing to care about encoding. * bugfix: ** building sphinx fails when package is not installed ** issues loading pretrained word2vec GoogleNews-vectors-negative300.bin (#1, #4)

3.3 v0.1

- First release.
- Supports word2vec and glove.
- Documentation using Sphinx.
- CLI interface for converting formats.

3.3.1 word_embedding_loader package

3.3.1.1 Subpackages

word_embedding_loader.loader package

loader module provides actual implementation of the file loaders.

Warning: This is an internal implementation. API may change without notice in the future, so you should use *word_embedding_loader.word_embedding.WordEmbedding*

Submodules

word_embedding_loader.loader.glove module

Low level API for loading of word embedding file that was implemented in GloVe, Global Vectors for Word Representation, by Jeffrey Pennington, Richard Socher, Christopher D. Manning from Stanford NLP group.

word_embedding_loader.loader.glove.**check_valid**(*line0*, *line1*)

Check if a file is valid Glove format.

Parameters

- **line0** (*bytes*) – First line of the file
- **line1** (*bytes*) – Second line of the file

Returns `True` if it is valid. `False` if it is invalid.

Return type boo

word_embedding_loader.loader.glove.**load**(*fin*, *dtype=<type 'numpy.float32'>*, *max_vocab=None*)

Load word embedding file.

Parameters

- **fin** (*File*) – File object to read. File should be open for reading ascii.
- **dtype** (*numpy.dtype*) – Element data type to use for the array.
- **max_vocab** (*int*) – Number of vocabulary to read.

Returns Word embedding representation vectors dict: Mapping from words to vector indices.

Return type numpy.ndarray

word_embedding_loader.loader.glove.**load_with_vocab**(*fin*, *vocab*, *dtype=<type 'numpy.float32'>*)

Load word embedding file with predefined vocabulary

Parameters

- **fin** (*File*) – File object to read. File should be open for reading ascii.
- **vocab** (*dict*) – Mapping from words (`bytes`) to vector indices (`int`).
- **dtype** (*numpy.dtype*) – Element data type to use for the array.

Returns Word embedding representation vectors

Return type numpy.ndarray

word_embedding_loader.loader.vocab module

word_embedding_loader.loader.vocab.**load_vocab**(*fin*)
Load vocabulary from vocab file created by word2vec with `-save-vocab <file>` option.

Parameters

- **fin** (*File*) – File-like object to read from.
- **encoding** (*bytes*) – Encoding of the input file as defined in `codecs` module of Python standard library.
- **errors** (*bytes*) – Set the error handling scheme. The default error handler is 'strict' meaning that encoding errors raise ValueError. Refer to `codecs` module for more information.

Returns

Mapping from a word (bytes) to the number of appearance in the original text (`int`). Order are preserved from the original vocab file.

Return type OrderedDict

word_embedding_loader.loader.word2vec_bin module

Low level API for loading of word embedding file that was implemented in word2vec, by Mikolov. This implementation is for word embedding file created with `-binary 1` option.

word_embedding_loader.loader.word2vec_bin.**check_valid**()
Check *word_embedding_loader.loader.glove.check_valid()* for the API.

word_embedding_loader.loader.word2vec_bin.**load**()
Refer to *word_embedding_loader.loader.glove.load()* for the API.

word_embedding_loader.loader.word2vec_bin.**load_with_vocab**()
Refer to *word_embedding_loader.loader.glove.load_with_vocab()* for the API.

word_embedding_loader.loader.word2vec_text module

Low level API for loading of word embedding file that was implemented in word2vec, by Mikolov. This implementation is for word embedding file created with `-binary 0` option (the default).

word_embedding_loader.loader.word2vec_text.**check_valid**(*line0*, *line1*)
Check *word_embedding_loader.loader.glove.check_valid()* for the API.

word_embedding_loader.loader.word2vec_text.**load**(*fin*, *dtype=<type 'numpy.float32'>*, *max_vocab=None*)
Refer to *word_embedding_loader.loader.glove.load()* for the API.

word_embedding_loader.loader.word2vec_text.**load_with_vocab**(*fin*, *vocab*, *dtype=<type 'numpy.float32'>*)
Refer to *word_embedding_loader.loader.glove.load_with_vocab()* for the API.

word_embedding_loader.saver package

loader module provides actual implementation of the file savers.

Warning: This is an internal implementation. API may change without notice in the future, so you should use *word_embedding_loader.word_embedding.WordEmbedding*

Submodules

word_embedding_loader.saver.glove module

Low level API for saving of word embedding file that was implemented in GloVe, Global Vectors for Word Representation, by Jeffrey Pennington, Richard Socher, Christopher D. Manning from Stanford NLP group.

word_embedding_loader.saver.glove.**save**(*f*, *arr*, *vocab*)
Save word embedding file.

Parameters

- **f** (*File*) – File to write the vectors. File should be open for writing ascii.
- **arr** (*numpy.array*) – Numpy array with float dtype.
- **vocab** (*iterable*) – Each element is pair of a word (bytes) and arr index (int). Word should be encoded to str apriori.

word_embedding_loader.saver.word2vec_bin module

Low level API for loading of word embedding file that was implemented in word2vec, by Mikolov. This implementation is for word embedding file created with -binary 1 option.

word_embedding_loader.saver.word2vec_bin.**save**()
Refer to *word_embedding_loader.saver.glove.save()* for the API.

word_embedding_loader.saver.word2vec_text module

Low level API for saving of word embedding file that was implemented in word2vec, by Mikolov. This implementation is for word embedding file created with -binary 0 option (the default).

word_embedding_loader.saver.word2vec_text.**save**(*f*, *arr*, *vocab*)
Save word embedding file. Check *word_embedding_loader.saver.glove.save()* for the API.

3.3.1.2 Submodules

3.3.1.3 word_embedding_loader.cli module

3.3.1.4 word_embedding_loader.exceptions module

exception word_embedding_loader.exceptions.**ParseError**
Bases: exceptions.Exception

exception word_embedding_loader.exceptions.**ParseWarning**
Bases: exceptions.Warning

word_embedding_loader.exceptions.**parse_warn**(*message*)

3.3.1.5 word_embedding_loader.word_embedding module

class word_embedding_loader.word_embedding.**WordEmbedding**(*vectors*, *vocab*, *freqs=None*)
Bases: object

Main API for loading and saving of pretrained word embedding files.

Note: You do not need to call initializer directly in normal usage. Instead you should call *load()*.

Parameters

- **vectors** (*numpy.ndarray*) – Word embedding representation vectors
- **vocab** (*dict*) – Mapping from words (bytes) to vector indices (int).
- **freqs** (*dict*) – Mapping from words (bytes) to word frequency counts (int).

vectors
numpy.ndarray – Word embedding vectors in shape of (vocabulary size, feature dimension).

vocab
dict – Mapping from words (bytes) to vector indices (int)

freqs
dict or None – Mapping from words (bytes) to frequency counts (int).

classmethod load(*path*, *vocab=None*, *dtype=<type 'numpy.float32'>*, *max_vocab=None*, *format=None*, *binary=False*)
Load pretrained word embedding from a file.

Parameters

- **path** (*str*) – Path of file to load.
- **vocab** (*str or None*) – Path to vocabulary file created by word2vec with -save-vocab <file> option. If vocab is given, *vectors* and *vocab* is ordered in descending order of frequency.
- **dtype** (*numpy.dtype*) – Element data type to use for the array.
- **max_vocab** (*int*) – Number of vocabulary to read.
- **format** (*str or None*) – Format of the file. 'word2vec' for file that was implemented in word2vec, by Mikolov et al.. 'glove' for file that was implemented in GloVe, Global Vectors for Word Representation, by Jeffrey Pennington, Richard Socher, Christopher D. Manning from Stanford NLP group. If None is given, the format is guessed from the content.
- **binary** (*bool*) – Load file as binary file as in word embedding file created by word2vec with -binary 1 option. If format is 'glove' or None, this argument is simply ignored

Returns *WordEmbedding*

save(*path*, *format*, *binary=False*, *use_load_condition=False*)
Save object as word embedding file. For most arguments, you should refer to *load()*.

Parameters **use_load_condition** (*bool*) – If *True*, options from *load()* is used.

Raises ValueError – use_load_condition == True but the object is not initialized via *load()*.

size
Feature dimension of the loaded vector.

Returns int

word_embedding_loader.word_embedding.**classify_format**(*f*)
Determine the format of word embedding file by their content. This operation only looks at the first two lines and does not check the sanity of input file.

Parameters **f** (*Filelike*) –

Returns class

CHAPTER 4

Indices and tables

- genindex
- modindex
- search

Python Module Index

W

Index