glove_word_vectors

September 29, 2021

1 Download Pre-Trained Global Vectors (GloVe) for Word Representation

We use a dataset that contains 1.6 million training and 350 test tweets from 2009 with algorithmically assigned binary positive and negative sentiment scores that are fairly evenly split.

1.1 Imports & Settings

[1]:
```
from pathlib import Path
import requests
from io import BytesIO
from zipfile import ZipFile
from tqdm import tqdm
```

1.2 Download and unzip

You can learn more about the data and manually download them from here.

[2]:
```
path = Path('glove')
if not path.exists():
    path.mkdir()
```

[3]:
```
URLs = ['http://nlp.stanford.edu/data/glove.6B.zip',
        'http://nlp.stanford.edu/data/glove.twitter.27B.zip',
        'http://nlp.stanford.edu/data/glove.840B.300d.zip']
```

[4]:
```
all_targets = [('glove.6B.100d.txt', 'glove.6B.300d.txt'),
               ('glove.twitter.27B.200d.txt',),
               ('glove.840B.300d.txt',)]
```

Downloads can take 10-20 min per target or longer, dpending on your connection. You can paste one of the urls in a browser do check download speed & time estimate.

[5]:
```
for url, targets in zip(URLs, all_targets):
    print(f'downloading {targets}...')
    response = requests.get(url).content
    print('done')
    with ZipFile(BytesIO(response)) as zip_file:
```

```
        for file in tqdm(zip_file.namelist()):
            if file in targets:
                local_file = path / file
                if not local_file.exists():
                    with local_file.open('wb') as output:
                        for line in zip_file.open(file).readlines():
                            output.write(line)
```

```
downloading ('glove.6B.100d.txt', 'glove.6B.300d.txt')…
  0%|          | 0/4 [00:00<?, ?it/s]
done
100%|      | 4/4 [00:11<00:00,  2.85s/it]
downloading ('glove.twitter.27B.200d.txt',)…
  0%|          | 0/4 [00:00<?, ?it/s]
done
100%|      | 4/4 [00:16<00:00,  4.02s/it]
downloading ('glove.840B.300d.txt',)…
  0%|          | 0/1 [00:00<?, ?it/s]
done
100%|      | 1/1 [00:58<00:00, 58.75s/it]
```