Simple GloVe word embeddings in R

B Kleinberg

02/09/2019

EURO CSS 2019 Workshop on Word Embeddings for Computational Social Science

Simple Glove word embeddings locally in R

The code below will show how to initialise and run a GloVe model of your choice on your local machine in R.

Downloading pretrained GloVe models

You need to download the GloVe models from https://nlp.stanford.edu/projects/glove/, unzip them and organise them in the following directory tree structure:

- glove
  - glove.6B.50d.txt
  - glove.6B.100d.txt
  - etc.

RAM requirements

This code is written so that you can use GloVe models on local machines with small memory. Ideal is 16GB but lower dimensional models work with 8GB. Note that the initialisation will take a few minutes.

Wrapper functions

Two functions wrap the initialisation and preparation of GloVe for R:

1. `setup_pipe(dir)`, which checks whether you have pointed R to the correct directory with GloVe models in it (= it tests whether you have pointed R to the folder with the downloaded models)
2. `init_glove(dir, which_model, dimensions)`, which initialises the desired GloVe model as a Document-Feature-Matrix object from the quanteda package (this facilitates *fast* vector similarity calculations).

You can access these function on GitHub at https://github.com/ben-aaron188/r_helper_functions/blob/master/init_glove.R

Initialising models locally

1. Either use the code locally or call the functions above from the local source (as done below - this requires that the `init_glove.R` script is located in the same folder as this R notebook):

```
source('./init_glove.R')
```

2. Initialise a Glove model

Here you can choose the size of the model (i.e. which model from the 6B, 42B, etc.) and the number of dimensions.

For example, for the 6B model with 100 dimensions, you would call:

```
init_glove(dir = './glove', which_model = '6B', dim=100)
```

This will load the model and print the progress and messages to your console:

```
[1] "Looking for pretrained GloVe vectors in: /Users/bennettkleinberg/Documents/glove"
[1] "Success - found GloVe objects in directory."
[1] "--- initialising the 100d model ---"
[1] "Success: initialised GloVe model as glove.pt"
```

3. Usse the model:

- Calcuate vector neighbours based on cosine distance for "cat" and "man" (using the `textstat_simil()` function from quanteda)

```
cos_sim_vals = textstat_simil(glove.pt
                              , selection = c("man", "cat")
                              , margin = "documents"
                              , method = "cosine")
```

- Show the Top 10 neighbours for "man"

```
head(sort(cos_sim_vals[,1], decreasing = TRUE), 10)
```

Output similar to:

```
      man     woman       boy       one    person   another       old
1.0000000 0.8323494 0.7914871 0.7788749 0.7526816 0.7522236 0.7409117
     life    father    turned
0.7371697 0.7370323 0.7347695
```

- Show the Top 10 neighbours for "cat"

```
head(sort(cos_sim_vals[,2], decreasing = TRUE), 10)
```

Output similar to:

```
      cat       dog    rabbit      cats    monkey       pet      dogs
1.0000000 0.8798075 0.7424427 0.7323004 0.7288710 0.7190140 0.7163873
    mouse     puppy       rat
0.6915251 0.6800068 0.6641027
```

- Same as above with another GloVe model and hence different vector distances:

```
init_glove(dir = './glove', which_model = '42B', dim=300)

cos_sim_vals = textstat_simil(glove.pt
                              , selection = c("man", "cat")
                              , margin = "documents"
                              , method = "cosine")
```

- Output 42B, 300d model for "man":

```
head(sort(cos_sim_vals[,1], decreasing = TRUE), 10)
```

```
      man     woman       guy        he       boy       men       him
1.0000000 0.8047993 0.7209722 0.7086842 0.6997220 0.6888777 0.6603094
      one    person       who
0.6583484 0.6545628 0.6502959
```

and "cat":

```r
head(sort(cos_sim_vals[,2], decreasing = TRUE), 10)
```

```
      cat      cats       dog    kitten       pet     kitty      dogs
1.0000000 0.7989762 0.7885448 0.7550500 0.7330315 0.6883491 0.6766946
    puppy    animal   kittens
0.6289975 0.6256891 0.6217528
```
