Configuration

```
#Parameters
PROJECT_NAME = 'ML1010_Weekly'
ENABLE_COLAB = True

#Root Machine Learning Directory. Projects appear underneath
GOOGLE_DRIVE_MOUNT = '/content/gdrive'
COLAB_ROOT_DIR = GOOGLE_DRIVE_MOUNT + '/MyDrive/Colab Notebooks'
COLAB_INIT_DIR = COLAB_ROOT_DIR + '/utility_files'

LOCAL_ROOT_DIR = '/home/magni/Documents/ML_Projects'
LOCAL_INIT_DIR = LOCAL_ROOT_DIR + '/utility_files'
```

Bootstrap Environment

```
#add in support for utility file directory and importing
import sys
import os

if ENABLE_COLAB:
  #Need access to drive
  from google.colab import drive
  drive.mount(GOOGLE_DRIVE_MOUNT, force_remount=True)

  #add in utility directory to syspath to import
  INIT_DIR = COLAB_INIT_DIR
  sys.path.append(os.path.abspath(INIT_DIR))

  #Config environment variables
  ROOT_DIR = COLAB_ROOT_DIR

else:
  #add in utility directory to syspath to import
  INIT_DIR = LOCAL_INIT_DIR
  sys.path.append(os.path.abspath(INIT_DIR))

  #Config environment variables
  ROOT_DIR = LOCAL_ROOT_DIR

#Import Utility Support
from jarvis import Jarvis
jarvis = Jarvis(ROOT_DIR, PROJECT_NAME)

import my python utils as myutils
```

```
Mounted at /content/gdrive
Wha...where am I?
I am awake now.
I am inspecting the local environment...
Your environment has been configured:
PROJECT_NAME:      ML1010_Weekly
ROOT_DIR: /content/gdrive/MyDrive/Colab Notebooks
ROOT_DATA_DIR: /content/gdrive/MyDrive/Colab Notebooks
DATA_DIR:     /content/gdrive/MyDrive/Colab Notebooks/data/ML1010_Weekly
WORKING_DIR:  /content/gdrive/MyDrive/Colab Notebooks/ML1010_Weekly
UTILITY_DIR:  /content/gdrive/MyDrive/Colab Notebooks/utility_files

Here are all your project work files
[D] /content/gdrive/MyDrive/Colab Notebooks/ML1010_Weekly
---[ipynb]-----> _template_wkX_ML1010_.ipynb (5.41 KB)
---[ipynb]-----> wk0_CSML1010_Day1.ipynb (27.77 KB)
---[ipynb]-----> wk0_Sentiment Analysis Tutorial.ipynb (28.92 KB)
---[ipynb]-----> wk1_ML1010_Code1 (1).ipynb (32.46 KB)
---[ipynb]-----> wk1_ML1010_Code1.ipynb (24.37 KB)
---[  pdf]-----> wk1_ML1010_Code1and2.pdf (1.44 MB)
---[ipynb]-----> wk1_ML1010_Code2.ipynb (1.84 MB)
---[ipynb]-----> wk1_text_classification_rnn.ipynb (16.92 KB)
---[ipynb]-----> wk2_ML1010_Code_FE.ipynb (58.81 KB)
---[ipynb]-----> wk2_ML1010_Flair_tutorial_3.ipynb (47.66 KB)
---[ipynb]-----> wk2_final_bert_long_docs_yay.ipynb (159.04 KB)

[D] /content/gdrive/MyDrive/Colab Notebooks/ML1010_Weekly/wk2_attempts
---[ipynb]-----> wk2_final_bert_long_docs.ipynb (140.11 KB)
---[ipynb]-----> wk2_final_bert_long_docs_v2.ipynb (140.17 KB)
---[ipynb]-----> wk2_final_bert_long_docs_v3.ipynb (138.70 KB)

Here are all your project data files
[D] /content/gdrive/MyDrive/Colab Notebooks/data/ML1010_Weekly
---[  gz][  csv]--> complaints.csv.gz (370.67 MB)
[*][  csv]-----> movie_reviews_cleaned.csv (38.37 MB)
---[  gz][  tsv]--> rspct.tsv.gz (347.13 MB)
---[  gz][  csv]--> subreddit_info.csv.gz (37.80 KB)

[D] /content/gdrive/MyDrive/Colab Notebooks/data/ML1010_Weekly/01_original
----->** No files **

[D] /content/gdrive/MyDrive/Colab Notebooks/data/ML1010_Weekly/02_working
----->** No files **

[D] /content/gdrive/MyDrive/Colab Notebooks/data/ML1010_Weekly/03_train
----->** No files **

[D] /content/gdrive/MyDrive/Colab Notebooks/data/ML1010_Weekly/04_test
----->** No files **

I have set your current working directory to /content/gdrive/MyDrive/Colab Notebooks/ML
The current time is 10:38
Hello sir. Extra caffeine may help.
```

Setup Runtime Environment

```
if ENABLE_COLAB:
  #!pip install scipy -q
  #!pip install scikit-learn -q
  #!pip install pycaret -q
  #!pip install matplotlib -q
  #!pip install joblib -q
  #!pip install pandasql -q


  print('Google Colab enabled')
else:
  print('Google Colab not enabled')


#Common imports
```

```
    Google Colab enabled
```

Flair Tutorial 3

```
!pip install git+https://github.com/flairNLP/flair.git
```

```
    Collecting git+https://github.com/flairNLP/flair.git
      Cloning https://github.com/flairNLP/flair.git to /tmp/pip-req-build-ws0zgyp6
      Running command git clone -q https://github.com/flairNLP/flair.git /tmp/pip-req-bui
      Installing build dependencies ... done
      Getting requirements to build wheel ... done
        Preparing wheel metadata ... done
    Requirement already satisfied: ftfy in /usr/local/lib/python3.7/dist-packages (from f
    Requirement already satisfied: gensim>=3.4.0 in /usr/local/lib/python3.7/dist-package
    Requirement already satisfied: torch!=1.8,>=1.5.0 in /usr/local/lib/python3.7/dist-pa
    Requirement already satisfied: regex in /usr/local/lib/python3.7/dist-packages (from
    Requirement already satisfied: wikipedia-api in /usr/local/lib/python3.7/dist-package
    Requirement already satisfied: lxml in /usr/local/lib/python3.7/dist-packages (from f
    Requirement already satisfied: transformers>=4.0.0 in /usr/local/lib/python3.7/dist-p
    Requirement already satisfied: scikit-learn>=0.21.3 in /usr/local/lib/python3.7/dist-
    Requirement already satisfied: more-itertools~=8.8.0 in /usr/local/lib/python3.7/dist
    Requirement already satisfied: konoha<5.0.0,>=4.0.0 in /usr/local/lib/python3.7/dist-
    Requirement already satisfied: tqdm>=4.26.0 in /usr/local/lib/python3.7/dist-packages
    Requirement already satisfied: tabulate in /usr/local/lib/python3.7/dist-packages (fr
    Requirement already satisfied: matplotlib>=2.2.3 in /usr/local/lib/python3.7/dist-pac
    Requirement already satisfied: gdown==3.12.2 in /usr/local/lib/python3.7/dist-package
    Requirement already satisfied: bpemb>=0.3.2 in /usr/local/lib/python3.7/dist-packages
    Requirement already satisfied: mpld3==0.3 in /usr/local/lib/python3.7/dist-packages (
    Requirement already satisfied: langdetect in /usr/local/lib/python3.7/dist-packages (
    Requirement already satisfied: conllu>=4.0 in /usr/local/lib/python3.7/dist-packages
```

```
Requirement already satisfied: huggingface-hub in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: sentencepiece==0.1.95 in /usr/local/lib/python3.7/dist
Requirement already satisfied: segtok>=1.5.7 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: janome in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: deprecated>=1.2.4 in /usr/local/lib/python3.7/dist-pac
Requirement already satisfied: sqlitedict>=1.6.0 in /usr/local/lib/python3.7/dist-pac
Requirement already satisfied: python-dateutil>=2.6.1 in /usr/local/lib/python3.7/dis
Requirement already satisfied: six in /usr/local/lib/python3.7/dist-packages (from gd
Requirement already satisfied: requests[socks] in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: filelock in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: wrap<2,>=1.10 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: scipy>=0.18.1 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: smart-open>=1.2.1 in /usr/local/lib/python3.7/dist-pac
Requirement already satisfied: overrides<4.0.0,>=3.0.0 in /usr/local/lib/python3.7/di
Requirement already satisfied: importlib-metadata<4.0.0,>=3.7.0 in /usr/local/lib/pyt
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: typing-extensions>=3.6.4 in /usr/local/lib/python3.7/d
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-pac
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: charset-normalizer~=2.0.0 in /usr/local/lib/python3.7/
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-pa
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/python3.7/dist
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: tokenizers<0.11,>=0.10.1 in /usr/local/lib/python3.7/d
Requirement already satisfied: sacremoses in /usr/local/lib/python3.7/dist-packages (
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: wcwidth in /usr/local/lib/python3.7/dist-packages (fro
```

```python
from flair.embeddings import WordEmbeddings
from flair.data import Sentence

glove_embedding = WordEmbeddings('glove')


sentence = Sentence("I am the very model of a modern major general.")

glove_embedding.embed(sentence)

for token in sentence:
  print(token)
  print(token.embedding)
```

```
Token: 1 I
tensor([-0.0465,  0.6197,  0.5665, -0.4658, -1.1890,  0.4460,  0.0660,  0.3191,
         0.1468, -0.2212,  0.7924,  0.2991,  0.1607,  0.0253,  0.1868, -0.3100,
        -0.2811,  0.6051, -1.0654,  0.5248,  0.0642,  1.0358, -0.4078, -0.3801,
         0.3080,  0.5996, -0.2699, -0.7603,  0.9422, -0.4692, -0.1828,  0.9065,
         0.7967,  0.2482,  0.2571,  0.6232, -0.4477,  0.6536,  0.7690, -0.5123,
```

```
                -0.4433, -0.2187,  0.3837, -1.1483, -0.9440, -0.1506,  0.3001, -0.5781,
                 0.2017, -1.6591, -0.0792,  0.0264,  0.2205,  0.9971, -0.5754, -2.7266,
                 0.3145,  0.7052,  1.4381,  0.9913,  0.1398,  1.3474, -1.1753,  0.0040,
                 1.0298,  0.0646,  0.9089,  0.8287, -0.4700, -0.1058,  0.5916, -0.4221,
                 0.5733, -0.5411,  0.1077,  0.3978, -0.0487,  0.0646, -0.6144, -0.2860,
                 0.5067, -0.4976, -0.8157,  0.1641, -1.9630, -0.2669, -0.3759, -0.9585,
                -0.8584, -0.7158, -0.3234, -0.4312,  0.4139,  0.2837, -0.7093,  0.1500,
                -0.2154, -0.3762, -0.0325,  0.8062])
    Token: 2 am
    tensor([-1.4022e-01,  2.4659e-02,  1.5813e-01, -7.4253e-04, -1.2708e+00,
             5.9635e-01,  3.0512e-02,  4.3477e-01,  2.8873e-01, -3.3262e-01,
             1.6353e+00,  5.3879e-01,  4.7854e-01, -3.5306e-01, -4.0265e-01,
            -1.5931e+00, -4.8427e-01, -2.6284e-01, -4.2667e-01,  6.2364e-01,
             1.6457e-01,  1.1028e+00, -4.6606e-01, -5.3797e-01,  2.3573e-01,
             5.0096e-01,  1.0428e+00,  6.7830e-02,  7.9727e-01,  1.0218e+00,
            -4.5716e-01,  1.3106e+00,  4.0831e-01, -1.3620e-01, -6.4568e-02,
            -2.2818e-01, -5.6815e-01,  2.2408e-02,  6.8127e-01, -4.5810e-01,
            -3.9547e-01, -6.3511e-01,  4.1244e-01, -2.6057e-01, -6.2107e-01,
            -1.1814e+00,  8.3442e-01, -6.0578e-01, -4.3701e-01, -1.6099e+00,
            -6.1338e-01, -2.1641e-01,  6.6364e-01,  7.5882e-01, -1.1520e+00,
            -1.9441e+00,  3.2438e-01,  6.6302e-01,  1.1977e+00, -2.7462e-01,
             5.1451e-01,  9.6280e-01, -1.2008e+00, -7.3438e-01,  5.0494e-01,
            -2.2075e-01, -2.4546e-01,  2.5850e-01, -7.9854e-01, -2.1524e-01,
            -2.0451e-01, -7.0773e-01,  2.4116e-01, -5.6108e-01,  5.4264e-01,
             2.2784e-01,  9.7165e-01,  1.1062e-01, -8.4572e-01, -6.4369e-01,
            -8.5003e-02, -8.2998e-01, -2.0660e-01,  1.7701e-02, -6.7984e-01,
             2.4250e-01, -4.8939e-01,  2.2466e-01, -8.1469e-01, -2.6440e-01,
             1.8299e-01,  5.2239e-01,  7.9125e-01,  9.1315e-01, -4.3357e-01,
             1.2760e-01, -1.2119e-01, -6.4749e-01,  4.8866e-01,  1.5445e-01])
    Token: 3 the
    tensor([-0.0382, -0.2449,  0.7281, -0.3996,  0.0832,  0.0440, -0.3914,  0.3344,
            -0.5755,  0.0875,  0.2879, -0.0673,  0.3091, -0.2638, -0.1323, -0.2076,
             0.3340, -0.3385, -0.3174, -0.4834,  0.1464, -0.3730,  0.3458,  0.0520,
             0.4495, -0.4697,  0.0263, -0.5415, -0.1552, -0.1411, -0.0397,  0.2828,
             0.1439,  0.2346, -0.3102,  0.0862,  0.2040,  0.5262,  0.1716, -0.0824,
            -0.7179, -0.4153,  0.2033, -0.1276,  0.4137,  0.5519,  0.5791, -0.3348,
            -0.3656, -0.5486, -0.0629,  0.2658,  0.3020,  0.9977, -0.8048, -3.0243,
             0.0125, -0.3694,  2.2167,  0.7220, -0.2498,  0.9214,  0.0345,  0.4674,
             1.1079, -0.1936, -0.0746,  0.2335, -0.0521, -0.2204,  0.0572, -0.1581,
            -0.3080, -0.4162,  0.3797,  0.1501, -0.5321, -0.2055, -1.2526,  0.0716,
             0.7056,  0.4974, -0.4206,  0.2615, -1.5380, -0.3022, -0.0734, -0.2831,
             0.3710, -0.2522,  0.0162, -0.0171, -0.3898,  0.8742, -0.7257, -0.5106,
            -0.5203, -0.1459,  0.8278,  0.2706])
    Token: 4 very
    tensor([-0.8414,  0.3099,  0.0582, -0.1282, -0.5756, -0.0910, -0.1414,  0.2938,
            -0.1028, -0.3223, -0.1437, -0.1539,  0.2740, -0.4129, -0.2128, -0.4144,
            -0.5921, -0.1556,  0.0722,  0.5727,  0.3717,  0.2452,  0.1877, -0.3333,
             0.0464,  0.1892, -0.2269, -0.6037, -0.1936,  0.4576, -0.4707,  0.6637,
             0.0645, -0.2661,  0.8150,  0.4137, -0.0435,  0.2204, -0.1121, -0.3496,
            -0.1082, -0.0761,  0.3813, -0.3779, -0.2158, -0.3776,  0.9711,  0.4465,
            -0.1220, -0.9115,  0.3719, -0.4478,  0.3636,  0.9872, -0.3910, -2.8503,
```

```
german_embedding = WordEmbeddings('de-crawl')
```

Flair Tutorial 4

```
from flair.embeddings import FlairEmbeddings

# init embedding
flair_embedding_forward = FlairEmbeddings('news-forward')

# create a sentence
sentence = Sentence('There must be some kind of way out of here.')

# embed words in sentence
flair_embedding_forward.embed(sentence)
```

```
    [Sentence: "There must be some kind of way out of here ."    [- Tokens: 11]]
```

```
# init forward embedding for German
flair_embedding_forward = FlairEmbeddings('en-forward')
flair_embedding_backward = FlairEmbeddings('en-backward')


from flair.embeddings import WordEmbeddings, FlairEmbeddings

# init standard GloVe embedding
glove_embedding = WordEmbeddings('glove')

# init Flair forward and backwards embeddings
flair_embedding_forward = FlairEmbeddings('news-forward')
flair_embedding_backward = FlairEmbeddings('news-backward')

from flair.embeddings import StackedEmbeddings

# create a StackedEmbedding object that combines glove and forward/backward flair embeddings
stacked_embeddings = StackedEmbeddings([
                                        glove_embedding,
                                        flair_embedding_forward,
                                        flair_embedding_backward,
                                       ])


sentence = Sentence('Now instantiate the StackedEmbeddings class and pass it a list containir

# just embed a sentence using the StackedEmbedding as you would with any single embedding.
stacked_embeddings.embed(sentence)

# now check out the embedded tokens.
for token in sentence:
    print(token)
    print(token.embedding)
```

```
Token: 1 Now
tensor([-1.4495e-02,  5.9107e-01,  7.0469e-01,  ...,  8.6380e-06,
        -2.5329e-03, -7.6302e-03])
Token: 2 instantiate
tensor([-1.1246e-01,  6.5809e-02, -4.3011e-01,  ..., -2.0528e-02,
        -1.1072e-02, -1.0217e-04])
Token: 3 the
tensor([-0.0382, -0.2449,  0.7281,  ..., -0.0068, -0.0048,  0.0067])
Token: 4 StackedEmbeddings
tensor([ 0.0000e+00,  0.0000e+00,  0.0000e+00,  ..., -6.3635e-05,
        -2.2326e-02, -4.5483e-03])
Token: 5 class
tensor([ 0.4451,  1.1348,  0.3368,  ..., -0.0019, -0.0136, -0.0022])
Token: 6 and
tensor([-7.1953e-02,  2.3127e-01,  2.3731e-02,  ...,  1.9294e-04,
        -4.8390e-03,  7.0484e-03])
Token: 7 pass
tensor([ 4.8602e-02, -6.0629e-01,  5.9483e-01,  ...,  4.2839e-03,
        -7.0139e-03, -5.8117e-05])
Token: 8 it
tensor([-0.3066,  0.1682,  0.9851,  ..., -0.0040, -0.0028, -0.0530])
Token: 9 a
tensor([-0.2709,  0.0440, -0.0203,  ...,  0.0015, -0.0120,  0.0272])
Token: 10 list
tensor([-6.7604e-01,  4.6687e-01,  8.3063e-01,  ..., -2.8084e-05,
        -2.4046e-02, -3.6952e-03])
Token: 11 containing
tensor([-0.5756,  0.4825,  0.4478,  ..., -0.0010, -0.0093,  0.0349])
Token: 12 these
tensor([-0.6046,  0.5075, -0.1892,  ..., -0.0056, -0.0327,  0.0028])
Token: 13 two
tensor([-0.2015,  0.3274,  0.0005,  ..., -0.0005, -0.0245, -0.0027])
Token: 14 embeddings
tensor([ 0.7325,  0.3735,  0.0189,  ..., -0.0030, -0.0089,  0.0405])
Token: 15 .
tensor([-0.3398,  0.2094,  0.4635,  ...,  0.0005, -0.0177,  0.0032])
```

```python
from flair.embeddings import FlairEmbeddings, TransformerWordEmbeddings

# init Flair embeddings
flair_forward_embedding = FlairEmbeddings('multi-forward')
flair_backward_embedding = FlairEmbeddings('multi-backward')

# init multilingual BERT
bert_embedding = TransformerWordEmbeddings('bert-base-multilingual-cased')
```

```
2021-11-27 15:39:00,411 https://flair.informatik.hu-berlin.de/resources/embeddings/flai
100%|████████| 172513724/172513724 [00:04<00:00, 38350321.76B/s]2021-11-27 15:39:05,0

2021-11-27 15:39:05,302 removing temp file /tmp/tmpt1ucsj07
Downloading: 100%                                    29.0/29.0 [00:00<00:00, 748B/s]
```

```python
from flair.embeddings import StackedEmbeddings

# now create the StackedEmbedding object that combines all embeddings
stacked_embeddings = StackedEmbeddings(
    embeddings=[flair_forward_embedding, flair_backward_embedding, bert_embedding])


sentence = Sentence('The grass is green .')

# just embed a sentence using the StackedEmbedding as you would with any single embedding.
stacked_embeddings.embed(sentence)

# now check out the embedded tokens.
for token in sentence:
    print(token)
    print(token.embedding)
```

```
Token: 1 The
tensor([ 0.6800,  0.2429,  0.0012,  ...,  0.7343, -0.0732,  0.1896])
Token: 2 grass
tensor([ 2.9200e-01,  2.2066e-02,  4.5290e-05,  ...,  9.8494e-01,
        -5.7341e-01,  6.8034e-01])
Token: 3 is
tensor([-0.5447,  0.0229,  0.0078,  ..., -0.2840, -0.1061, -0.0851])
Token: 4 green
tensor([0.1477, 0.1097, 0.0009,  ..., 0.0203, 0.5680, 0.0867])
Token: 5 .
tensor([-1.5555e-01,  6.7599e-03,  5.3829e-06,  ..., -4.0763e-01,
         1.7429e-01,  3.1956e-02])
```

Flair Tutorial 5

```python
from flair.embeddings import WordEmbeddings, DocumentPoolEmbeddings

# initialize the word embeddings
glove_embedding = WordEmbeddings('glove')

# initialize the document embeddings, mode = mean
document_embeddings = DocumentPoolEmbeddings([glove_embedding])


# create an example sentence
sentence = Sentence('The grass is green . And the sky is blue .')
```

```
# embed the sentence with our document embedding
document_embeddings.embed(sentence)

# now check out the embedded sentence.
print(sentence.embedding)
```

```
tensor([-3.1970e-01,  2.6206e-01,  4.0371e-01, -4.8223e-01,  2.1118e-01,
         8.5380e-02, -6.0909e-02,  2.2149e-01, -2.4234e-01, -1.0128e-01,
         8.6213e-02, -1.6874e-01,  3.4736e-01,  6.7267e-02,  2.2750e-01,
        -2.5534e-01,  3.9017e-01,  9.6975e-03, -9.5909e-03,  2.8388e-02,
        -3.2033e-02, -5.7822e-03,  2.8569e-01,  4.0082e-02,  5.8185e-01,
         2.3183e-01,  5.9500e-02, -5.7468e-01, -2.0337e-01, -1.7826e-01,
        -1.8182e-01,  4.7222e-01,  9.8503e-02,  1.0854e-01,  1.9359e-01,
         2.9041e-01,  1.5739e-04,  4.3389e-01,  1.8119e-01, -1.1405e-01,
        -3.4222e-01, -4.9730e-01,  1.6268e-02, -1.8057e-01,  2.5105e-02,
         1.4868e-02,  2.3021e-01, -8.9935e-02, -4.4742e-02, -5.3620e-01,
        -1.3269e-01, -1.3503e-01,  2.4511e-01,  1.2051e+00, -4.5334e-01,
        -2.6632e+00,  2.7964e-02,  4.9859e-02,  1.5550e+00,  5.0574e-01,
        -8.0093e-02,  6.9114e-01, -1.5679e-01,  2.3944e-01,  9.0704e-01,
        -1.1536e-01,  3.8778e-01,  6.8844e-02,  3.2989e-01, -1.5260e-01,
        -4.2541e-02, -2.4333e-01, -1.6738e-01, -3.1495e-01,  2.4115e-01,
         6.8981e-02,  1.1922e-01,  1.4478e-01, -6.6563e-01,  6.9832e-02,
         5.8356e-01,  8.6527e-02, -4.8388e-01,  1.5086e-01, -8.9072e-01,
        -3.4816e-01, -2.6635e-02, -2.1770e-01,  3.8983e-01,  7.9085e-02,
        -2.3757e-02, -5.3694e-01, -3.1125e-01,  4.7508e-01, -6.1099e-01,
         9.1333e-02, -5.4229e-01, -2.9515e-01,  5.2432e-01,  1.9662e-01])
```

```
from flair.embeddings import WordEmbeddings, DocumentRNNEmbeddings

glove_embedding = WordEmbeddings('glove')

document_embeddings = DocumentRNNEmbeddings([glove_embedding])


# create an example sentence
sentence = Sentence('The grass is green . And the sky is blue .')

# embed the sentence with our document embedding
document_embeddings.embed(sentence)

# now check out the embedded sentence.
print(sentence.get_embedding())
```

```
tensor([ 0.0156,  0.4606, -0.4987,  0.1918, -0.2671,  0.1089, -0.1766,  0.0864,
        -0.2962,  0.1636, -0.1561,  0.1230, -0.1762, -0.1211,  0.4802, -0.4709,
         0.2924,  0.2896, -0.1228,  0.0573,  0.1639, -0.2966,  0.0922,  0.0702,
        -0.3568, -0.2863,  0.2676, -0.2769, -0.3739,  0.2383, -0.4473,  0.3044,
        -0.1232, -0.3890,  0.0055, -0.1390, -0.4262,  0.2078, -0.0984, -0.3085,
        -0.0439,  0.0394,  0.4127, -0.4268,  0.0615,  0.0860,  0.2795,  0.2232,
         0.2274, -0.3803, -0.0039,  0.5613,  0.2777,  0.0933, -0.2205, -0.0604,
        -0.1005,  0.0174,  0.0923,  0.3631, -0.4520,  0.0072,  0.1077,  0.1662,
         0.3450,  0.3121, -0.4186,  0.0603, -0.2659, -0.0476, -0.3909,  0.1460,
        -0.3344,  0.1134, -0.1357, -0.1878,  0.1246, -0.2921,  0.1624, -0.2554,
```

```
           0.2431, -0.3533, -0.1204, -0.1346, -0.0426, -0.0349,  0.0096, -0.5044,
           0.1857,  0.2479,  0.2630,  0.1047, -0.2688, -0.0413,  0.4856, -0.0806,
          -0.2258, -0.2726, -0.2558,  0.4438, -0.2989,  0.1974, -0.3665, -0.0633,
           0.4606,  0.2816,  0.1760, -0.1703,  0.2639,  0.3445,  0.1395,  0.2719,
           0.5012, -0.1031,  0.0673,  0.0510, -0.4296,  0.2119,  0.0733, -0.1397,
          -0.0157,  0.0731,  0.1207,  0.1317, -0.1886,  0.2532, -0.0507,  0.5494],
          grad_fn=<CatBackward0>)
```

```python
from flair.embeddings import TransformerDocumentEmbeddings

# init embedding
embedding = TransformerDocumentEmbeddings('bert-base-uncased')

# create a sentence
sentence = Sentence('The grass is green .')

# embed the sentence
embedding.embed(sentence)
```

Downloading: 100%	28.0/28.0 [00:00<00:00, 629B/s]
Downloading: 100%	570/570 [00:00<00:00, 11.8kB/s]
Downloading: 100%	226k/226k [00:00<00:00, 530kB/s]
Downloading: 100%	455k/455k [00:00<00:00, 626kB/s]
Downloading: 100%	420M/420M [00:10<00:00, 40.5MB/s]

```
[Sentence: "The grass is green ."   [− Tokens: 5]]
```

```python
from flair.embeddings import TransformerDocumentEmbeddings

# init embedding
embedding = TransformerDocumentEmbeddings('roberta-base')

# create a sentence
sentence = Sentence('The grass is green .')

# embed the sentence
embedding.embed(sentence)
```

Downloading: 670k/670k [00:00~00:00,

```
from flair.data import Sentence
from flair.embeddings import SentenceTransformerDocumentEmbeddings

# init embedding
#embedding = SentenceTransformerDocumentEmbeddings('bert-base-nli-mean-tokens')
embedding = SentenceTransformerDocumentEmbeddings('stsb-roberta-large')
# create a sentence
sentence = Sentence('The grass is green .')

# embed the sentence
embedding.embed(sentence)
```

```
2021-11-27 15:40:14,020 ------------------------------------------------------------
2021-11-27 15:40:14,022 ATTENTION! The library "sentence-transformers" is not inst
2021-11-27 15:40:14,025 To use Sentence Transformers, please first install with "p
2021-11-27 15:40:14,026 ------------------------------------------------------------
------------------------------------------------------------
UnboundLocalError                          Traceback (most recent call last)
<ipython-input-21-afed1b00f3b1> in <module>()
      4 # init embedding
      5 #embedding = SentenceTransformerDocumentEmbeddings('bert-base-nli-mean-
tokens')
----> 6 embedding = SentenceTransformerDocumentEmbeddings('stsb-roberta-large')
      7 # create a sentence
      8 sentence = Sentence('The grass is green .')

/usr/local/lib/python3.7/dist-packages/flair/embeddings/document.py in
__init__(self, model, batch_size, convert_to_numpy)
    735                 pass
    736
--> 737             self.model = SentenceTransformer(model)
    738             self.name = 'sentence-transformers-' + str(model)
    739             self.batch_size = batch_size

UnboundLocalError: local variable 'SentenceTransformer' referenced before
assignment
```