

ICEDEG TUTORIAL

SOCIAL MEDIA DATA FOR E-GOVERNMENT Digital Citizens and their Degree of Interest in Politics

LORENA RECALDE

(If you don't have IPython Notebook, please follow next steps. Otherwise, download the project from Github in and install the module named word2vec in your python environment. Please go to step 4)

1. Lounch JupyterLab to run (and do) some code. Open your web browser and go to next link. <https://mybinder.org/>

2. Copy next repository link in box <GitHub repo or URL>. Click on button <launch>

** WAIT few minutes until the project is uploaded.

Repository: https://github.com/lore10/ICEDEG_tutorial

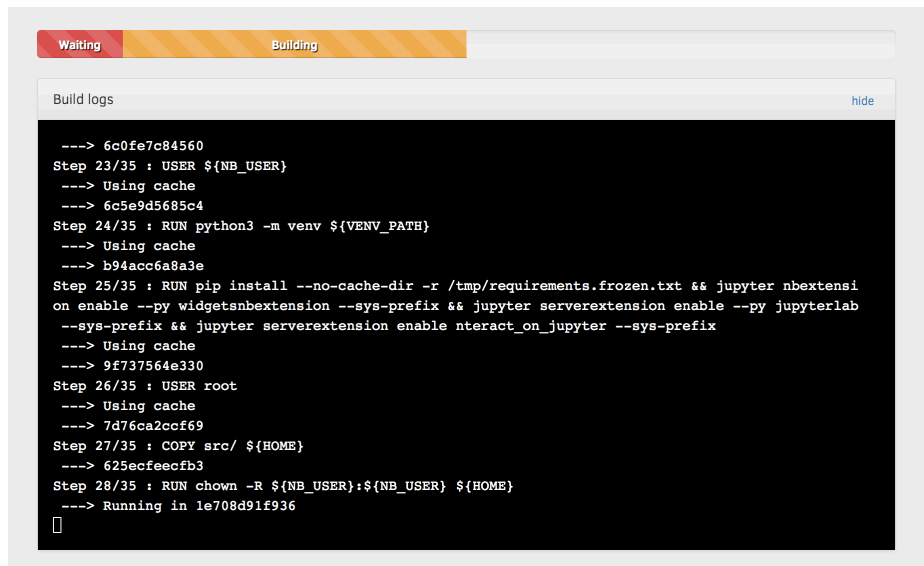


Turn a GitHub repo into a collection of
interactive notebooks

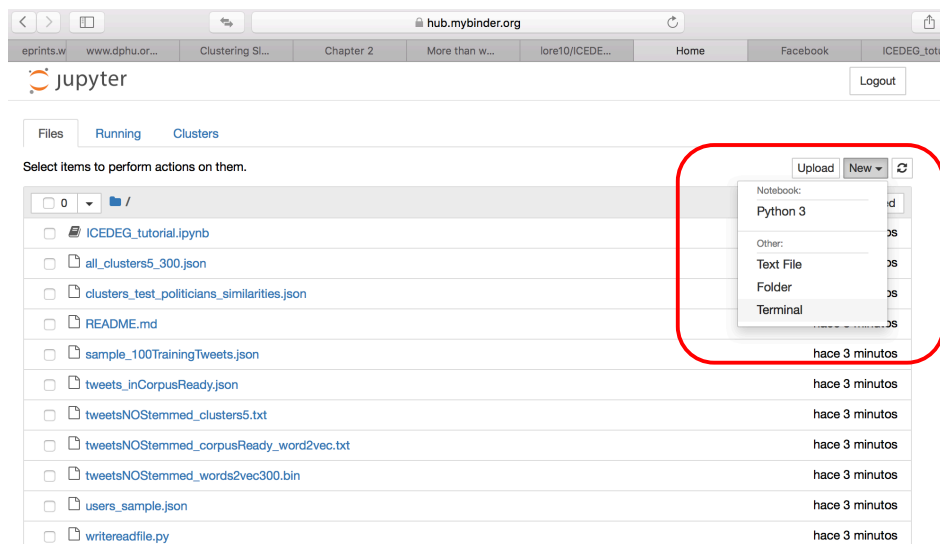
Have a repository full of Jupyter notebooks? With Binder, open those notebooks in an executable environment, making your code immediately reproducible by anyone, anywhere.

A screenshot of the Binder web interface. The title is "Build and launch a repository". There are two input fields: "GitHub repo or URL" containing "https://github.com/lore10/ICEDEG_tutorial" and "Git branch, tag, or commit" containing "master". To the right, there is a "Path to a notebook file (optional)" field and a "File" dropdown menu. A red box highlights the "GitHub repo or URL" field and the "launch" button. Below the input fields, there is a blue bar with the text "Copy the URL below and share your Binder with others:". At the bottom, there is a text box containing the URL "https://mybinder.org/v2/gh/lore10/ICEDEG_tutorial/master" and a copy icon.

While waiting you will see:

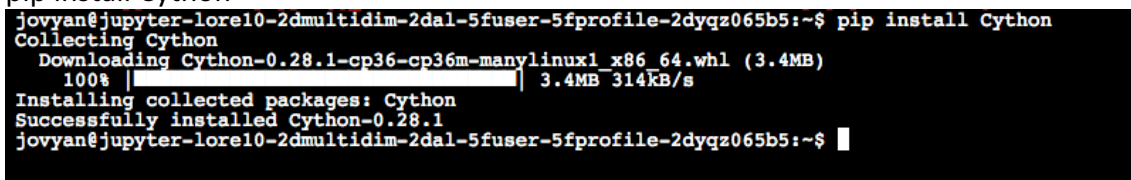


3. Go to New – Terminal. Install required packages

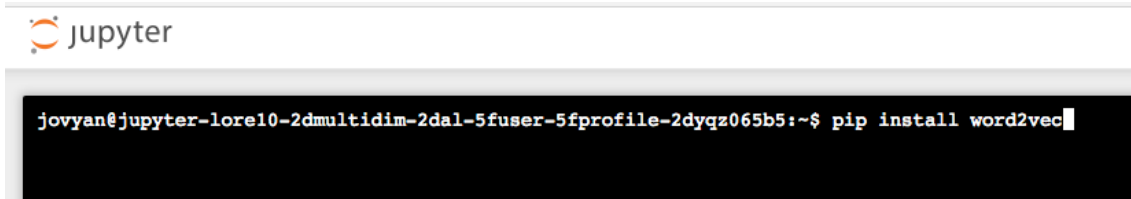


Follow next instructions:

pip install Cython



pip install word2vec



Install also:

```
pip install unidecode
```

```
pip install nltk
```

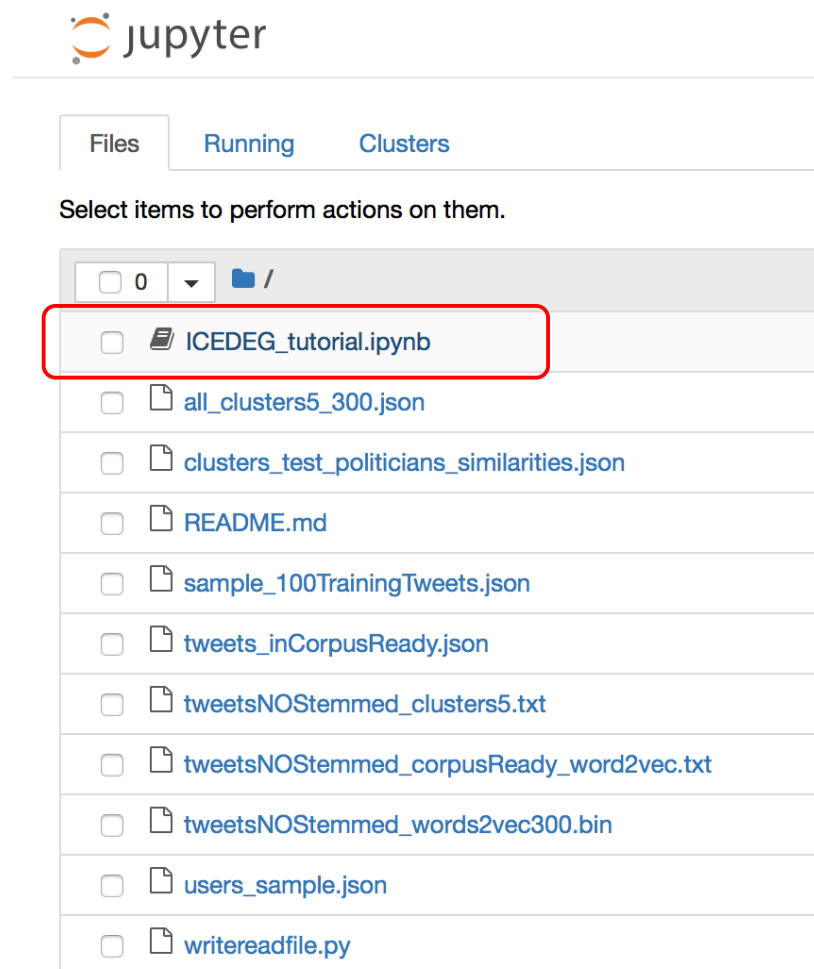
```
pip install stop_words
```

```
pip install sklearn
```

```
pip install scipy
```

```
pip install matplotlib
```

4. Open the file ICEDEG_tutorial.ipynb with one click on it.



PART 1

Training the word embeddings model, word2vec

5. See the decisions made about data cleaning/preprocessing.

6. Load the preprocessed tweets that were used to train the model. This is only a sample of 100 tweets. Here you have the original tweet and the result when it is preprocessed.

6. To train the model we need to provide as input a txt file with the training tweets (every tweet per line). This file is already prepared to run
>> word2vec.word2vec(input_file.txt, output_file.bin, OTHER PARAMETERS)

PART 2

Exploring the work of Word embeddings

- Do not forget to import the module word2vec and load the model
7. Follow item 5) in the notebook to see how similarities are found. This is useful to verify the quality of your trained model.
8. Follow item 6) and 7) in the notebook to see some plots in a 2D space.
9. Follow item 8) to work with analogies and item 9) to find how similar two words are.

PART 3

Clustering words in the vocabulary

10. Explore how the words were grouped and if the clusters make sense. How subjective may clustering be? Remember that in our proposal we worked with 5 clusters.

PART 4

How similar is a tweet to the Politics-related centroid?

11. Try some code to classify the given tweets into the corresponding cluster

PART 5

DoIP already calculated for some users

12. How the tweets of politicians are classified?

THANKS...