Natural Language Programming with Python

Fall 2021

Housekeeping

- In case of technical problems:
  - Something wrong on my end (e.g. power outage), I will send you an email.
  - Something wrong on your end, please send me a text message. 508-769-6446
  - jcodygroup@gmail.com
- We have 4 hours for each session
  - I will try to give you an opportunity to stand and stretch every hour.
  - We will take at least one 15-minute break near the halfway point.

About me

- **Experience:**
  - 25+ years consulting and training experience
  - Extensive work with “big data” and analytics
  - 15 years working with various data visualization tools
- **Education**
  - Ed. M., Technology, Innovation & Education, Harvard University
  - PhD Candidate, Education Policy, University of Massachusetts, Amherst

NLP

What is Natural Language Processing (NLP)?

NLP is a broad field, encompassing a variety of tasks, including:

- Part-of-speech tagging: identify if each word is a noun, verb, adjective, etc.)
- Named entity recognition (NER): identify person names, organizations, locations, medical codes, etc
- Question answering
- Speech recognition
- Text-to-speech and Speech-to-text
- Topic modeling
- Sentiment classification
- Text Classification
- Text Generation
- Language modeling
- Translation

I like to organize it this way…

What group does this text belong to?	What is in this text?	What can I say about this text?

• Text classification
  • spam/not spam
  • finance/health/IT/etc.
  • semantic analysis
	• Named Entity Recognition
	• Topic Modeling
		• Text Summarization
		• Text Generation
		• Q & A

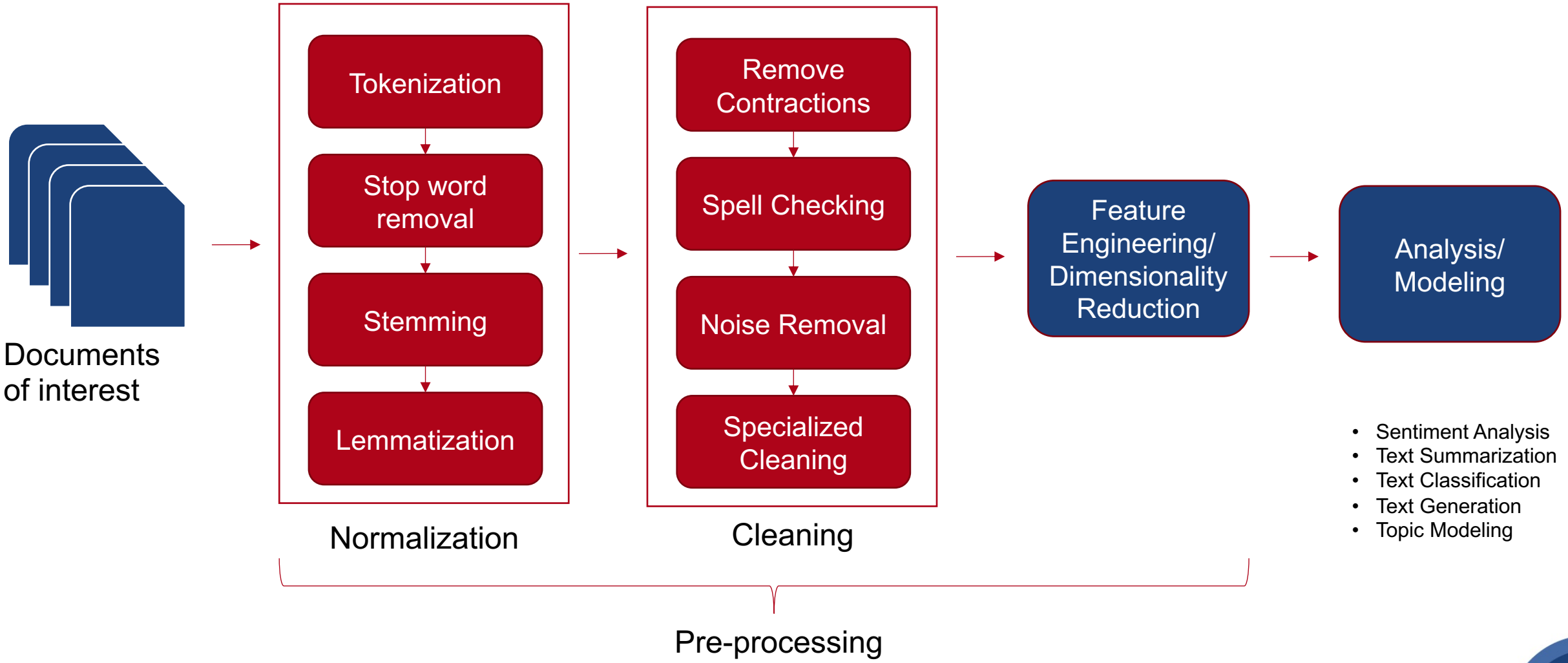
I have not included translation, speech-to-text, etc.

Our focus

NLP is a broad field, encompassing a variety of tasks, including:

- Part-of-speech tagging: identify if each word is a noun, verb, adjective, etc.)
- Named entity recognition (NER): identify person names, organizations, locations, medical codes, etc
- Question answering
- Speech recognition
- Text-to-speech and Speech-to-text
- **Topic modeling**
- **Sentiment classification**
- **Text Classification**
- **Text Generation**
- Language modeling
- Translation

General Process Flow



- Sentiment Analysis
- Text Summarization
- Text Classification
- Text Generation
- Topic Modeling

Is NLP part of Machine Learning?

Yes.

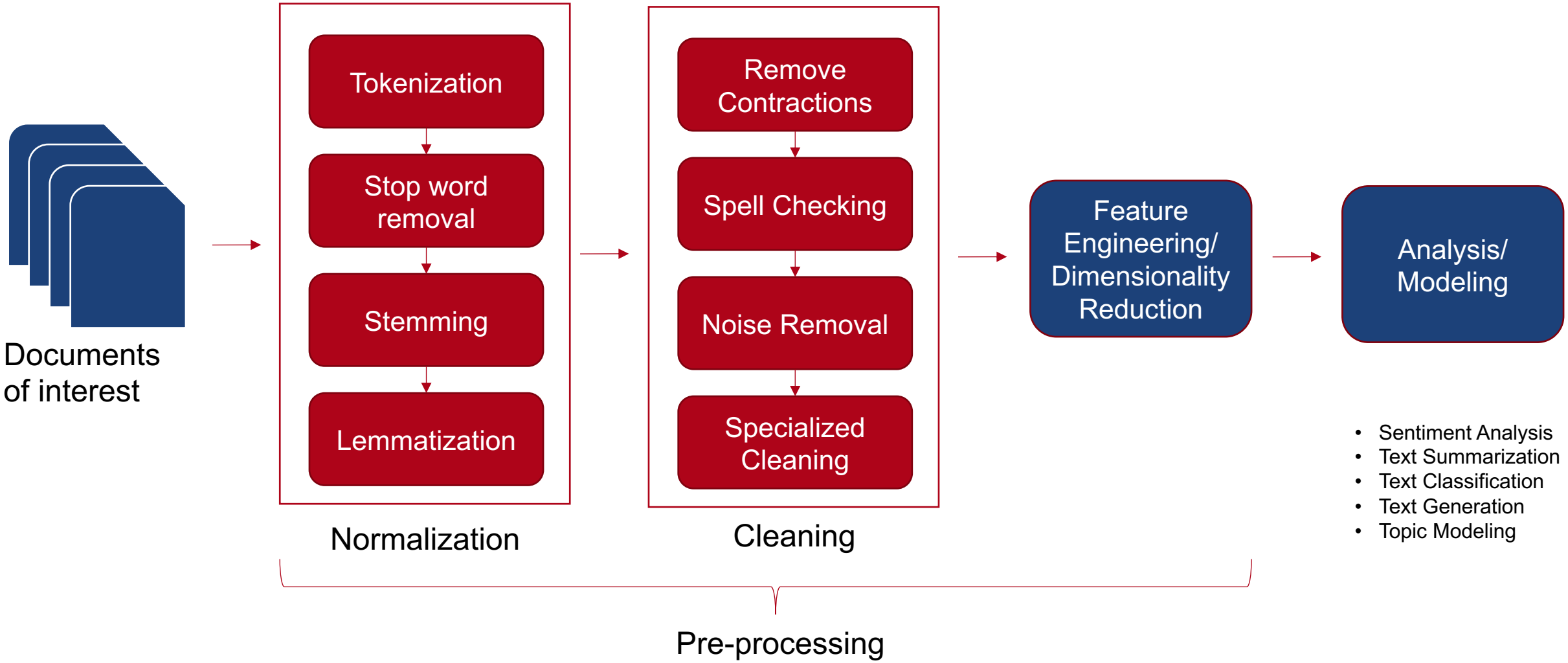
Working in Google CoLab

Patient comments about prescribed medications

Our data set contains comments from patients about medications they have been taking.

Can we determine the sentiment (pos, neg, neutral) of new comments based on this data?

General Process Flow



- Sentiment Analysis
- Text Summarization
- Text Classification
- Text Generation
- Topic Modeling

Terminology

- Algorithm: A set of rules/instructions/steps to follow in order to build a model.
- Model: A mathematical equation that produces the desired output
- NLP ‘Classifiers’ are algorithms
  - Naive Bayes Classifier
  - Linear Classifier
  - Support Vector Machine
  - Bagging Models
  - Boosting Models
  - Shallow Neural Networks
  - Deep Neural Networks
    - Convolutional Neural Network (CNN)
    - Long Short Term Modeler (LSTM)
    - Gated Recurrent Unit (GRU)
    - Bidirectional RNN
    - Recurrent Convolutional Neural Network (RCNN)
    - Other Variants of Deep Neural Networks

Machine Learning algorithms

- Supervised
  - regression
- Unsupervised
  - clustering

2 Training-based

In [44]:
```
training = [
('Iron man is the best.', 'pos'),
('Thor should have gone for the head', 'neg'),
('Hawkeye is the best Avenger', 'pos'),
('None of the Fantastic Four movies are good', 'neg'),
('Chris Evans is boring.', 'neg'),
('Age of Ultron was the most exciting Marvel movie', 'pos'),
('Well known actors impeded immersiveness of Marvel movies', 'neg'),
]
testing = [
('Superman was never an interesting character.','neg'),
('Fantastic Mr Fox is an awesome film!','pos'),
('Dragonball Evolution is simply terrible!!','neg')
]
```

In [45]:
```
from textblob import classifiers

# Naive Bayes classifier
classifier = classifiers.NaiveBayesClassifier(training)
```

In [47]:
```
print (classifier.accuracy(testing))
classifier.show_informative_features(3)
```

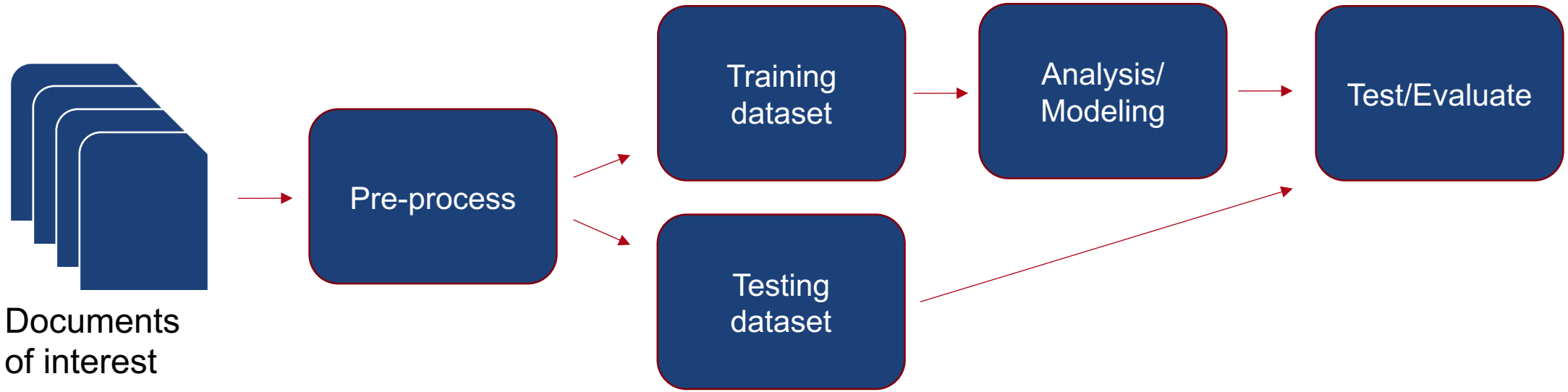
```
0.6666666666666666
Most Informative Features
          contains(best) = False             neg : pos    =      2.4 : 1.0
            contains(is) = True              pos : neg    =      2.1 : 1.0
            contains(is) = False             neg : pos    =      1.9 : 1.0
```

Based on the training data provided, if the statement contains the word 'is', there is a high probability that the text has negative sentiment.

In [46]:
```
blob = TextBlob('CDC is the best', classifier=classifier)
print (blob.classify())
```

```
pos
```

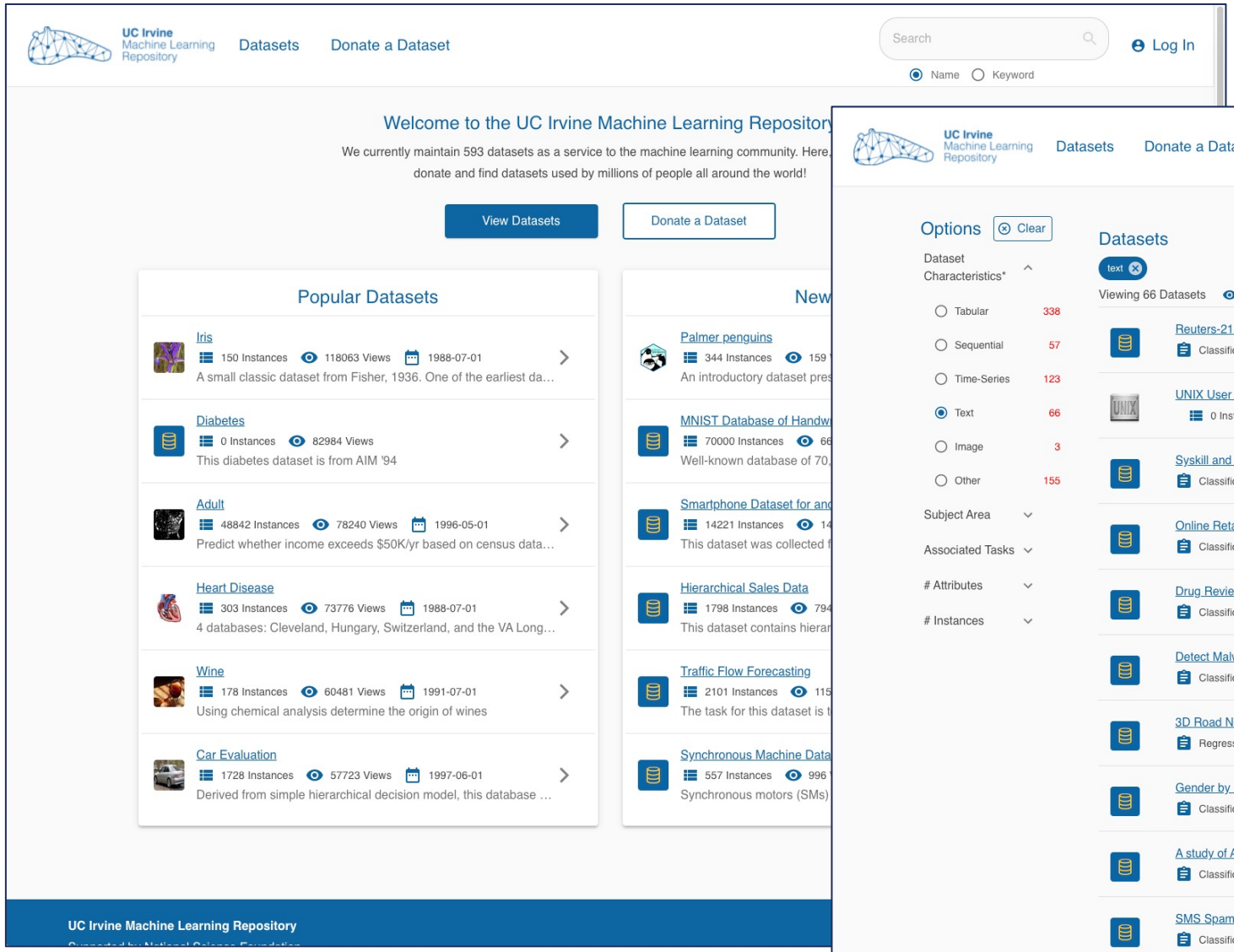
General Process Flow

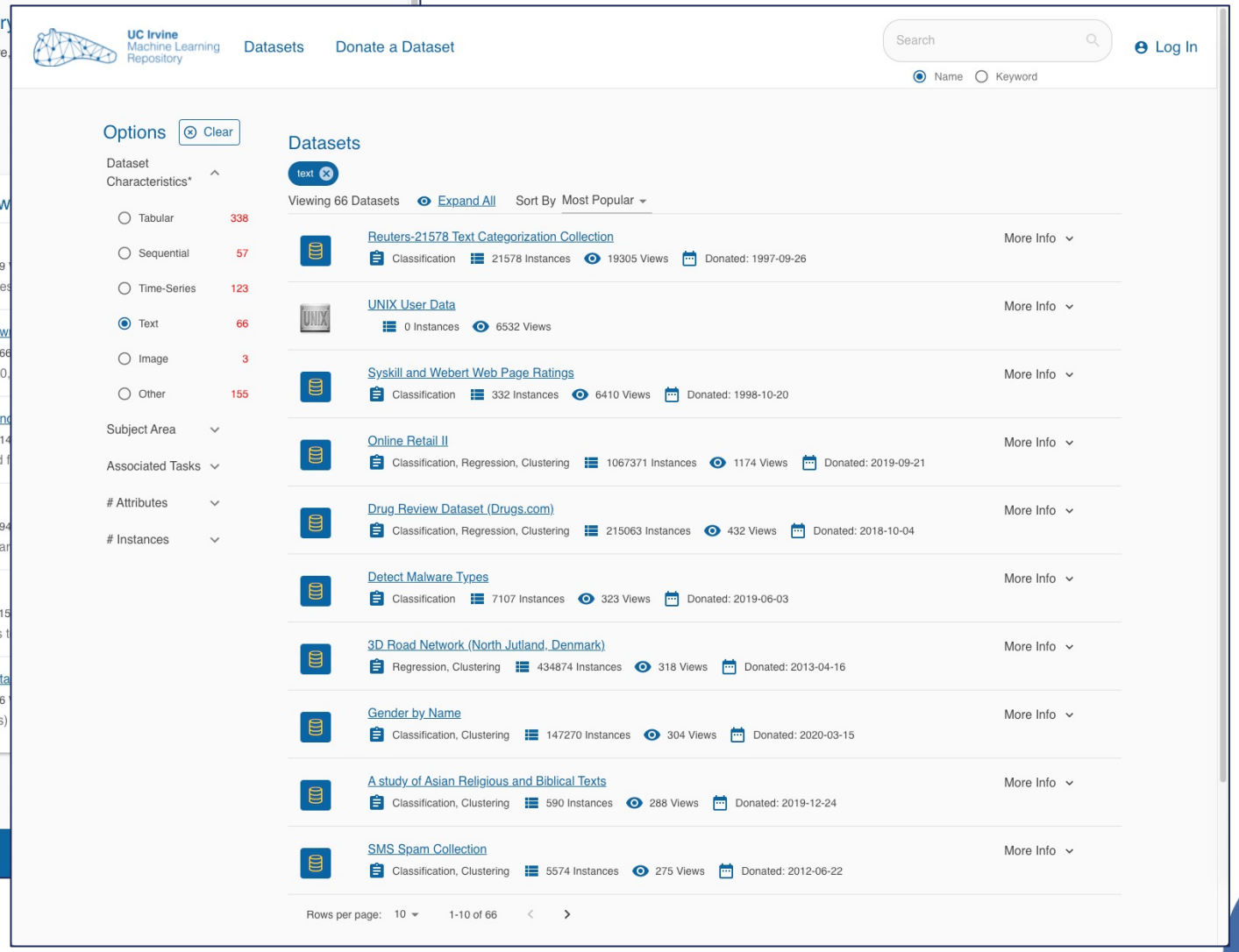


Where can we get data?

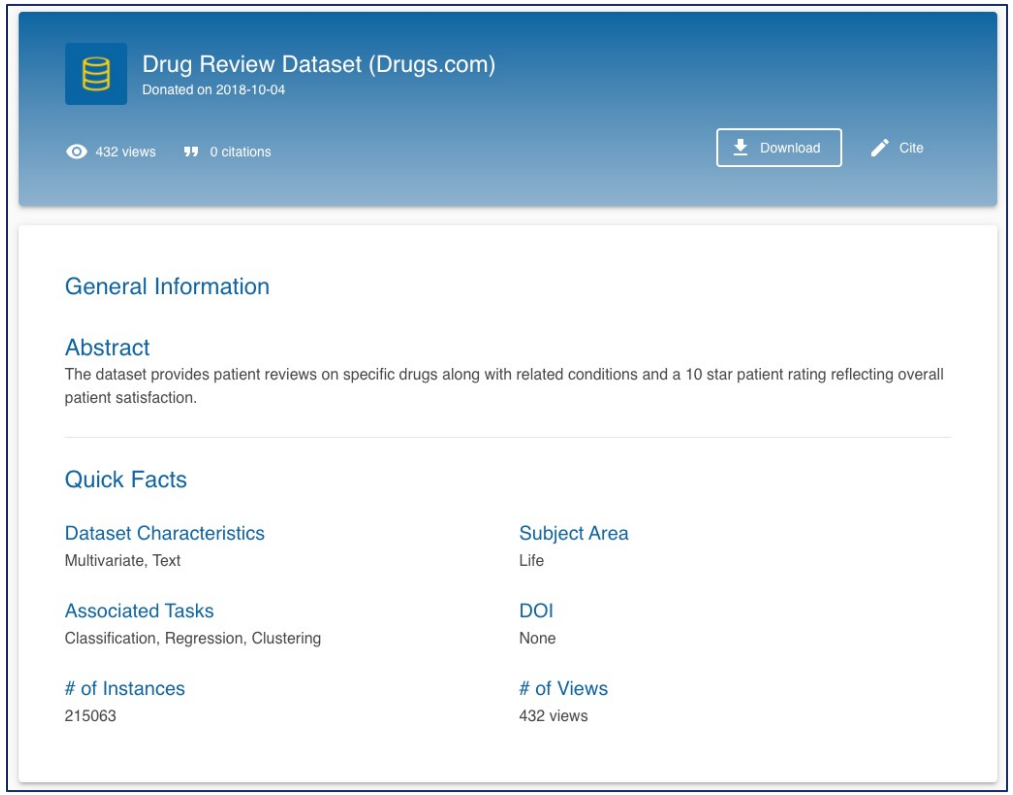
UCI data

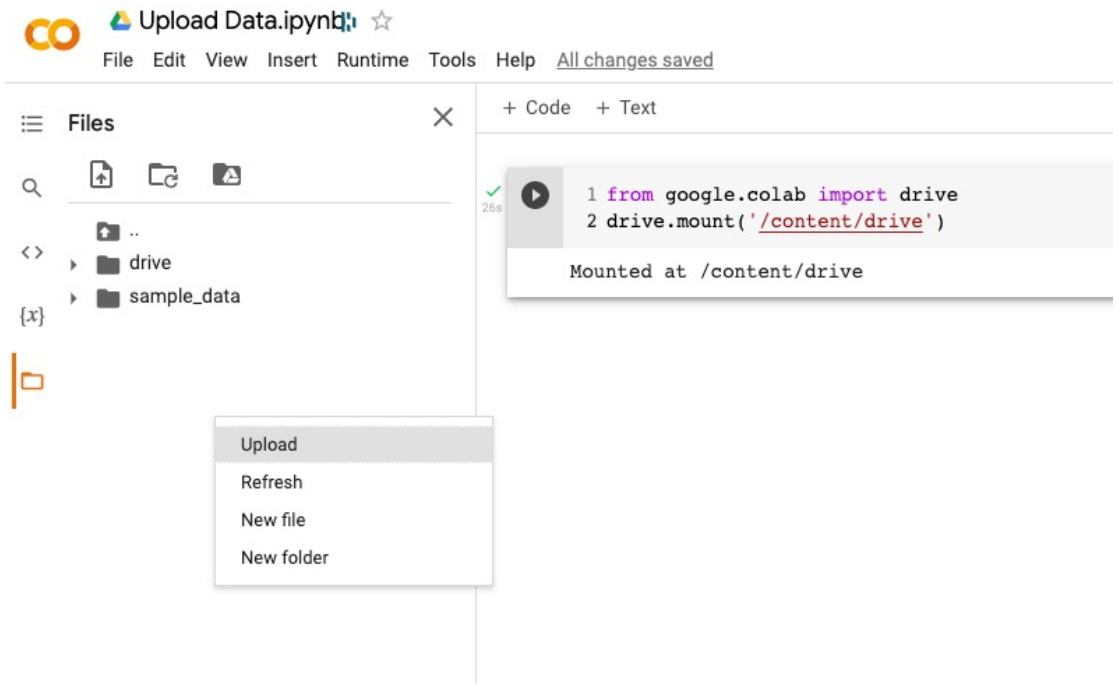
UCI Machine Learning repository



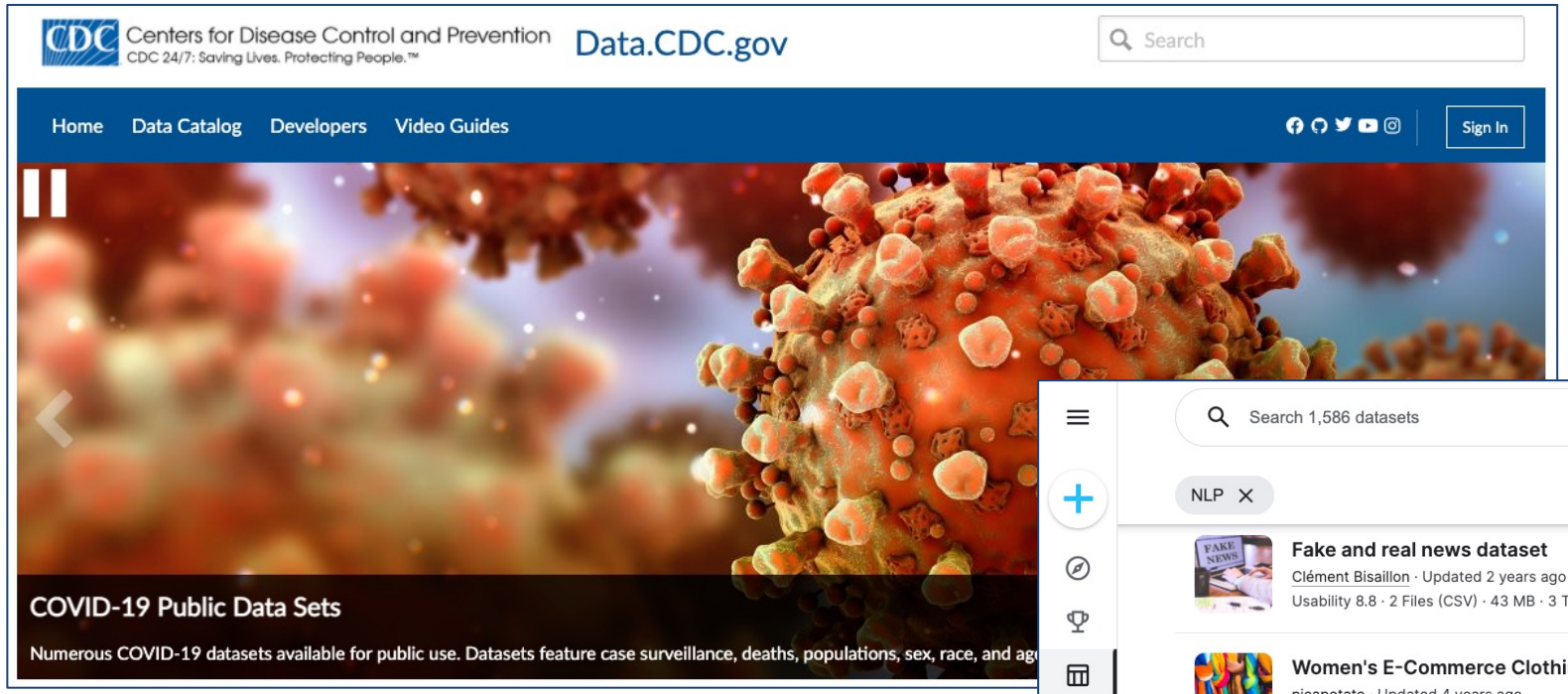


UCI data

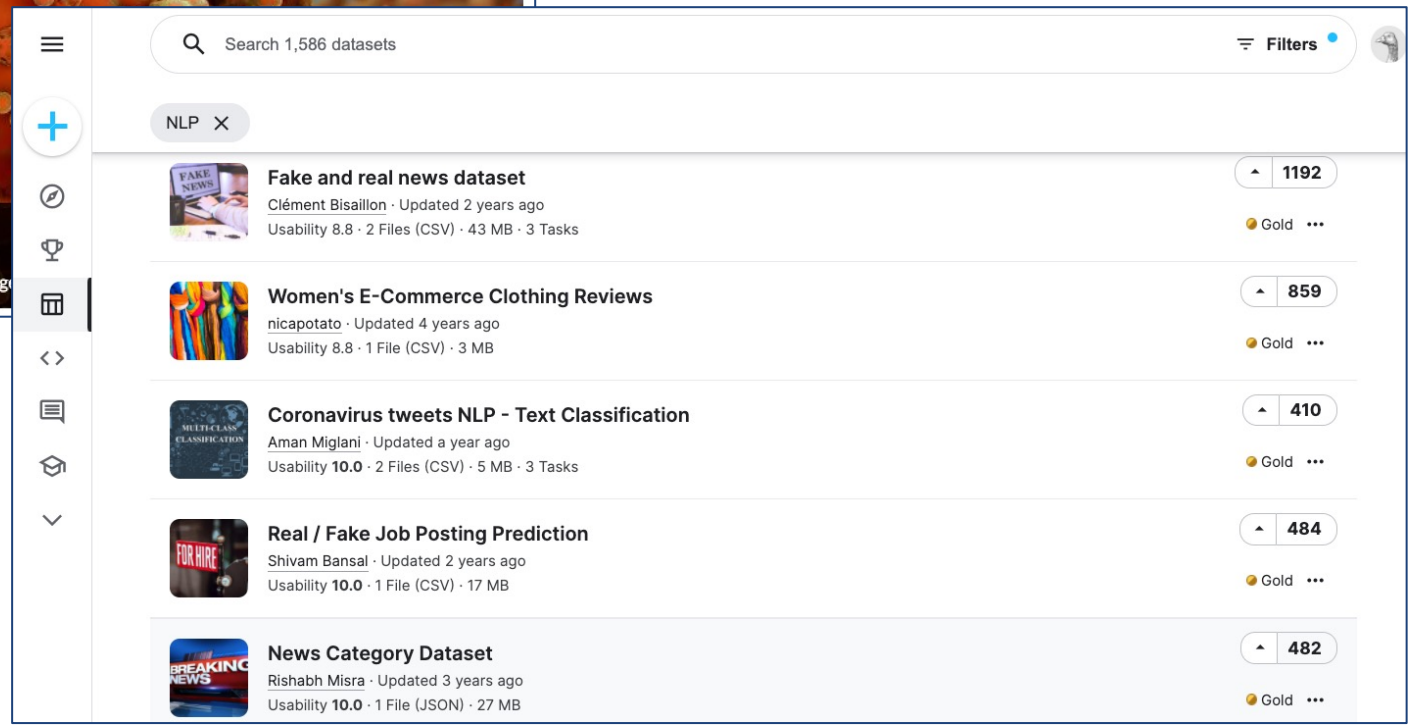

Drug Review Dataset (Drugs.com)
Donated on 2018-10-04
432 views
0 citations
Download
Cite
General Information
Abstract
The dataset provides patient reviews on specific drugs along with related conditions and a 10 star patient rating reflecting overall patient satisfaction.
Quick Facts
Dataset Characteristics
Multivariate, Text
Subject Area
Life
Associated Tasks
Classification, Regression, Clustering
DOI
None
# of Instances
215063
# of Views
432 views


Upload Data.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Files
+ Code + Text
..
drive
sample_data
1 from google.colab import drive
2 drive.mount('/content/drive')
Mounted at /content/drive
Upload
Refresh
New file
New folder

Sources of data



Kaggle



Pre-processing data

1 – Pre-processing

Exercise - Preprocessing

Use the Exercise – Preprocessing notebook as a starter

Instructions

1. Load this data set from kaggle - kaggle datasets download -d gpreda/pfizer-vaccine-tweets
2. Determine the shape of the dataframe
3. Review the data types
4. Drop the id column
5. Check for null values
6. Perform the following pre-processing on the 'text' column.
    - (new column1) change all text to lowercase
    - (new column2) use new column1 and remove contractions.
    - (new column3) use new column2 and string the data back together
    - (new column4) use new column3 and tokenize into sentences
    - (new column5) use new column3, again, and tokenize into words
    - (new column6) use new column5 and special characters
    - (new column7) use new column6 and remove stop words
    - (new column8) use new column7 and perform stemming
    - (new column9) use new column8 and perform lemmanization
    - add columns tweet length and tweet word count

EDA with Text

2 – EDA with Text: Data with labels

We are moving slightly beyond EDA in this notebook

- Feature engineering
- Vectorization
- Corpus
- Document Matrix

	word 1	word 2	word 3	word 4	word 5	word 6	word 7	word 8
phrase 1	0	1	0	0	0	0	1	1
phrase 2	1	1	0	0	0	1	0	1
phrase 3	0	0	1	1	1	0	1	0

Feature Engineering

- Raw text data will be transformed into feature vectors and new features will be created using the existing dataset.
- Bag of Words: all the words in the corpus without reference to what came before or after
- N-grams (beyond unigrams): sets of words can provide more context about the relationship between words.
  - CDC has a lot of scientist > ‘CDC has’, ‘has a’, ‘a lot’, ‘lot of’, ‘of scientist’
- Count vectors: matrix notation of the dataset in which every row represents a document from the corpus, every column represents a term from the corpus, and every cell represents the frequency count of a particular term in a particular document.
- TF-IDF vectors: represents the relative importance of a term in the document and the entire corpus.
  - TF-IDF score is composed by two terms: the first computes the normalized Term Frequency (TF), the second term is the Inverse Document Frequency (IDF), computed as the logarithm of the number of the documents in the corpus divided by the number of documents where the specific term appears.
- Word embedding - A word embedding is a form of representing words and documents using a dense vector representation. The position of a word within the vector space is learned from text and is based on the words that surround the word when it is used. Word embeddings can be trained using the input corpus itself or can be generated using pre-trained word embeddings such as **Glove, FastText,** and **Word2Vec**. Any one of them can be downloaded and used as transfer learning.
- Text/NLP features – counts of nouns, verbs, words, characters

Exercise – EDA with text

Use your Exercise – Preprocessing notebook

1. Add columns for polarity and subjectivity.
2. Create charts to show distributions of :
    1. polarity
    2. retweets
    3. user followers
    4. user friends
    5. favorites
    6. review length
    7. word count
    8. polarity by location
    9. retweets by location
3. After stop words have been removed, what are the top 10:
    1. unigrams
    2. bigrams
4. Add two other charts or tables you think might be interesting to see.