TextHero

Under the hoods, Texthero makes use of multiple NLP and machine learning toolkits such as Gensim, NLTK, SpaCy and scikit-learn. You don't need to install them all separately, pip will take care of that.

Texthero include tools for:

- Preprocess text data: it offers both out-of-the-box solutions but it's also flexible for custom-solutions.
- Natural Language Processing: keyphrases and keywords extraction, and named entity recognition.
- Text representation: TF-IDF, term frequency, and custom word-embeddings (wip)
- Vector space analysis: clustering (K-means, Meanshift, DBSAN and Hierarchical), topic modelling (wip) and interpretation.
- Text visualization: vector space visualization, place localization on maps (wip).

Supported representation algorithms:

- Term frequency (count)
- Term frequency-inverse document frequency (tfidf)

Supported clustering algorithms:

- K-means (kmeans)
- Density-Based Spatial Clustering of Applications with Noise (dbscan)
- Meanshift (meanshift)

Supported dimensionality reduction algorithms:

- Principal component analysis (pca)
- t-distributed stochastic neighbor embedding (tsne)
- Non-negative matrix factorization (nmf)

In [1]:
```
# !pip install texthero
```

In [2]:
```
import texthero
```

```
✓ Download and installation successful
You can now load the model via spacy.load('en_core_web_sm')
```

In [3]:
```
help(texthero)
```

```
Help on package texthero:

NAME
    texthero - Texthero: python toolkit for text preprocessing, representation and visualization.

PACKAGE CONTENTS
    nlp
    preprocessing
    representation
    stopwords
    visualization

DATA
    Callable = typing.Callable
    List = typing.List
    Optional = typing.Optional
    Set = typing.Set

FILE
    c:\users\panka\anaconda3\envs\nlp\lib\site-packages\texthero\__init__.py
```

Text Preprocessing

In [5]:
```
import pandas as pd
text="It's a pleasant   day at Bangaloré; at / (10:30) am"
series=pd.Series(text)
```

In [6]:
```
series
```

Out[6]:
```
0    It's a pleasant   day at Bangaloré; at / (10:3...
dtype: object
```

In [7]:
```
import texthero as hero

hero.remove_digits(series)
```

Out[7]:
```
0    It's a pleasant   day at Bangaloré; at / ( : ) am
dtype: object
```

In [8]:
```
#### Remove punctuations
hero.remove_punctuation(series)
```

Out[8]:
```
0    It s a pleasant   day at Bangaloré  at    10 3...
dtype: object
```

In [9]:
```
#### Remove Brackets
hero.remove_brackets(series)
```

Out[9]:
```
0    It's a pleasant   day at Bangaloré; at /  am
dtype: object
```

In [10]:
```
print(series)
hero.remove_diacritics(series)
```

```
0    It's a pleasant   day at Bangaloré; at / (10:3...
dtype: object
```

Out[10]:
```
0    It's a pleasant   day at Bangalore; at / (10:3...
dtype: object
```

In [11]:
```
hero.remove_whitespace(series)
```

Out[11]:
```
0    It's a pleasant day at Bangaloré; at / (10:30) am
dtype: object
```

In [12]:
```
### Stopwords
hero.remove_stopwords(series)
```

Out[12]:
```
0    It'  pleasant   day  Bangaloré;  / (10:30)
dtype: object
```

In [13]:
```
hero.clean(series)
```

Out[13]:
```
0    pleasant day bangalore
dtype: object
```

In [14]:
```
df = pd.read_csv(
   "https://github.com/jbesomi/texthero/raw/master/dataset/bbcsport.csv"
)
df.head()
```

Out[14]:

	text	topic
0	Claxton hunting first major medal\n\nBritish h...	athletics
1	O'Sullivan could run in Worlds\n\nSonia O'Sull...	athletics
2	Greene sets sights on world title\n\nMaurice G...	athletics
3	IAAF launches fight against drugs\n\nThe IAAF ...	athletics
4	Dibaba breaks 5,000m world record\n\nEthiopia'...	athletics

In [15]:
```
df.shape  # two columns
```

Out[15]:
```
(737, 2)
```

In [16]:
```
df['topic'].unique()
```

Out[16]:
```
array(['athletics', 'cricket', 'football', 'rugby', 'tennis'],
      dtype=object)
```

In [17]:
```
df['topic'].value_counts()
```

Out[17]:
```
football     265
rugby        147
cricket      124
athletics    101
tennis       100
Name: topic, dtype: int64
```

In [18]:
```
###PCA
import texthero as hero
import pandas as pd

# df = pd.read_csv(
#    "https://github.com/jbesomi/texthero/raw/master/dataset/bbcsport.csv"
# )

df['pca'] = (
   df['text']
   .pipe(hero.clean)
   .pipe(hero.tfidf)###vectorizing
   .pipe(hero.pca)
)
hero.scatterplot(df, 'pca', color='topic', title="PCA BBC Sport news")
```

In [63]:
```
df.head()
```

Out[63]:

	text	topic	pca
0	Claxton hunting first major medal\n\nBritish h...	athletics	[-0.09109912281144122, 0.10359351265238617]
1	O'Sullivan could run in Worlds\n\nSonia O'Sull...	athletics	[-0.00036132812221576415, 0.02478045501220412]
2	Greene sets sights on world title\n\nMaurice G...	athletics	[-0.11760496196780282, 0.12860286068425186]
3	IAAF launches fight against drugs\n\nThe IAAF ...	athletics	[-0.09134845338902024, 0.15398002814497108]
4	Dibaba breaks 5,000m world record\n\nEthiopia'...	athletics	[-0.0912957165291783, 0.13507109027104225]

In [2]:
```
df.head()
```

Out[2]:

	text	topic	pca
0	Claxton hunting first major medal\n\nBritish h...	athletics	[-0.09107819053003914, 0.10357210282741101]
1	O'Sullivan could run in Worlds\n\nSonia O'Sull...	athletics	[-0.0002786547625436705, 0.02477621330944455]
2	Greene sets sights on world title\n\nMaurice G...	athletics	[-0.11765703516162962, 0.12865601739827767]
3	IAAF launches fight against drugs\n\nThe IAAF ...	athletics	[-0.09131528756100005, 0.15397654273191513]
4	Dibaba breaks 5,000m world record\n\nEthiopia'...	athletics	[-0.0912807640539468, 0.13510622701055774]

In [ ]:
```

```

In [22]:
```
import texthero as hero
import pandas as pd

df = pd.read_csv(
    "https://github.com/jbesomi/texthero/raw/master/dataset/bbcsport.csv"
)

df['tfidf'] = (
    df['text']
    .pipe(hero.clean)
    .pipe(hero.tfidf)
)
```

In [23]:
```
df
```

Out[23]:

	text	topic	tfidf
0	Claxton hunting first major medal\n\nBritish h...	athletics	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
1	O'Sullivan could run in Worlds\n\nSonia O'Sull...	athletics	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
2	Greene sets sights on world title\n\nMaurice G...	athletics	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0533678197008...
3	IAAF launches fight against drugs\n\nThe IAAF ...	athletics	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
4	Dibaba breaks 5,000m world record\n\nEthiopia'...	athletics	[0.24734311047947527, 0.0, 0.0, 0.0, 0.0, 0.0,...
...	...	...	...
732	Agassi into second round in Dubai\n\nFourth se...	tennis	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
733	Mauresmo fights back to win title\n\nWorld num...	tennis	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
734	Federer wins title in Rotterdam\n\nWorld numbe...	tennis	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
735	GB players warned over security\n\nBritain's D...	tennis	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
736	Sharapova overcomes tough Molik\n\nWimbledon c...	tennis	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...

737 rows × 3 columns

In [35]:
```
df['tfidf'].head()
```

Out[35]:
```
0    [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
1    [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
2    [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0533678197008...
3    [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
4    [0.24734311047947527, 0.0, 0.0, 0.0, 0.0, 0.0,...
Name: tfidf, dtype: object
```

The default pipeline for the clean method is the following:

1. fillna(s) Replace not assigned values with empty spaces.\
2. lowercase(s) Lowercase all text.\
3. remove_digits() Remove all blocks of digits.\
4. remove_punctuation() Remove all string.punctuation (!"#$%&'()*+,-./:;<=>?@[]^_`{|}~).\
5. remove_diacritics() Remove all accents from strings.\
6. remove_stopwords() Remove all stop words.\
7. remove_whitespace() Remove all white space between words.\

In [36]:
```
from texthero import preprocessing

custom_pipeline = [preprocessing.fillna,
                   preprocessing.lowercase,
                   preprocessing.remove_whitespace]
df['clean_text'] = hero.clean(df['text'], custom_pipeline)
```

Before cleaning

In [37]:
```
df['text']
```

Out[37]:
```
0      Claxton hunting first major medal\n\nBritish h...
1      O'Sullivan could run in Worlds\n\nSonia O'Sull...
2      Greene sets sights on world title\n\nMaurice G...
3      IAAF launches fight against drugs\n\nThe IAAF ...
4      Dibaba breaks 5,000m world record\n\nEthiopia'...
                             ...
732    Agassi into second round in Dubai\n\nFourth se...
733    Mauresmo fights back to win title\n\nWorld num...
734    Federer wins title in Rotterdam\n\nWorld numbe...
735    GB players warned over security\n\nBritain's D...
736    Sharapova overcomes tough Molik\n\nWimbledon c...
Name: text, Length: 737, dtype: object
```

after cleaning

In [38]:
```
df['clean_text']
```

Out[38]:
```
0      claxton hunting first major medal british hurd...
1      o'sullivan could run in worlds sonia o'sulliva...
2      greene sets sights on world title maurice gree...
3      iaaf launches fight against drugs the iaaf - a...
4      dibaba breaks 5,000m world record ethiopia's t...
                             ...
732    agassi into second round in dubai fourth seed ...
733    mauresmo fights back to win title world number...
734    federer wins title in rotterdam world number o...
735    gb players warned over security britain's davi...
736    sharapova overcomes tough molik wimbledon cham...
Name: clean_text, Length: 737, dtype: object
```

In [43]:
```
# or alternatively
df['clean_text1'] = df['clean_text'].pipe(hero.clean, custom_pipeline)
```

In [44]:
```
df['clean_text1']
```

Out[44]:
```
0      claxton hunting first major medal british hurd...
1      o'sullivan could run in worlds sonia o'sulliva...
2      greene sets sights on world title maurice gree...
3      iaaf launches fight against drugs the iaaf - a...
4      dibaba breaks 5,000m world record ethiopia's t...
                             ...
732    agassi into second round in dubai fourth seed ...
733    mauresmo fights back to win title world number...
734    federer wins title in rotterdam world number o...
735    gb players warned over security britain's davi...
736    sharapova overcomes tough molik wimbledon cham...
Name: clean_text1, Length: 737, dtype: object
```

In [ ]:
```
Representation
Once cleaned the data, the next natural is to map each document into a vector.

TFIDF representation
```

In [45]:
```
df['tfidf_clean_text'] = hero.tfidf(df['clean_text'])
```

In [47]:
```
df.head()
```

Out[47]:

	text	topic	tfidf	clean_text	clean_text1	tfidf_clean_text
0	Claxton hunting first major medal\n\nBritish h...	athletics	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	claxton hunting first major medal british hurd...	claxton hunting first major medal british hurd...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
1	O'Sullivan could run in Worlds\n\nSonia O'Sull...	athletics	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	o'sullivan could run in worlds sonia o'sulliva...	o'sullivan could run in worlds sonia o'sulliva...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
2	Greene sets sights on world title\n\nMaurice G...	athletics	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0533678197008...	greene sets sights on world title maurice gree...	greene sets sights on world title maurice gree...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
3	IAAF launches fight against drugs\n\nThe IAAF ...	athletics	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	iaaf launches fight against drugs the iaaf - a...	iaaf launches fight against drugs the iaaf - a...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
4	Dibaba breaks 5,000m world record\n\nEthiopia'...	athletics	[0.24734311047947527, 0.0, 0.0, 0.0, 0.0, 0.0,...	dibaba breaks 5,000m world record ethiopia's t...	dibaba breaks 5,000m world record ethiopia's t...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...

In [48]:
```
df2 = df[['clean_text','tfidf_clean_text']]
```

In [49]:
```
# Clean text with out put feature
df2.head()
```

Out[49]:

	clean_text	tfidf_clean_text
0	claxton hunting first major medal british hurd...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
1	o'sullivan could run in worlds sonia o'sulliva...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
2	greene sets sights on world title maurice gree...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
3	iaaf launches fight against drugs the iaaf - a...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
4	dibaba breaks 5,000m world record ethiopia's t...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...

Dimensionality reduction with PCA

To visualize the data, we map each point to a two-dimensional representation with PCA. The principal component analysis algorithms returns the combination of attributes that better account the variance in the data.

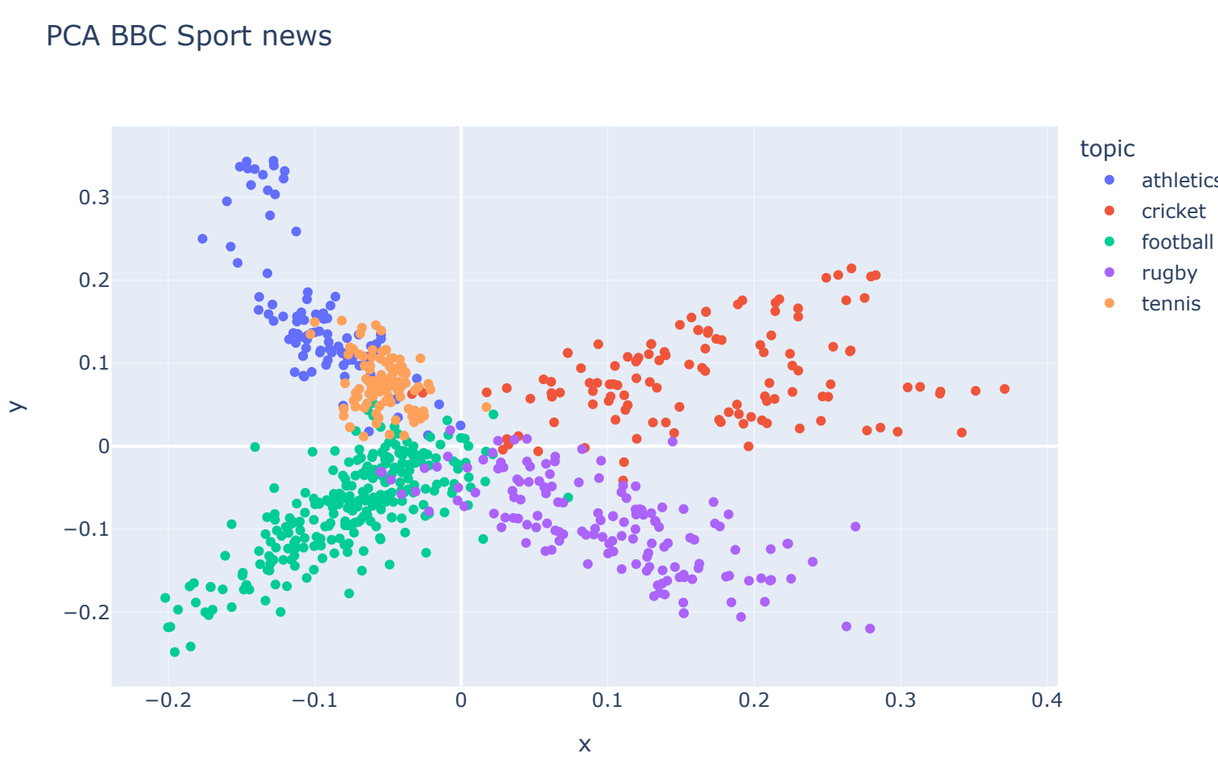
In [50]:
```
df['pca_tfidf_clean_text'] = hero.pca(df['tfidf_clean_text'])
```

In [51]:
```
df['pca'] = (
            df['text']
            .pipe(hero.clean)
            .pipe(hero.tfidf)
            .pipe(hero.pca)
   )
```

Visualization

texthero.visualization provide some helpers functions to visualize the transformed Dataframe. All visualization utilize under the hoods the Plotly Python Open Source Graphing Library.

In [53]:
```
hero.scatterplot(df, col='pca', color='topic', title="PCA BBC Sport news")
```



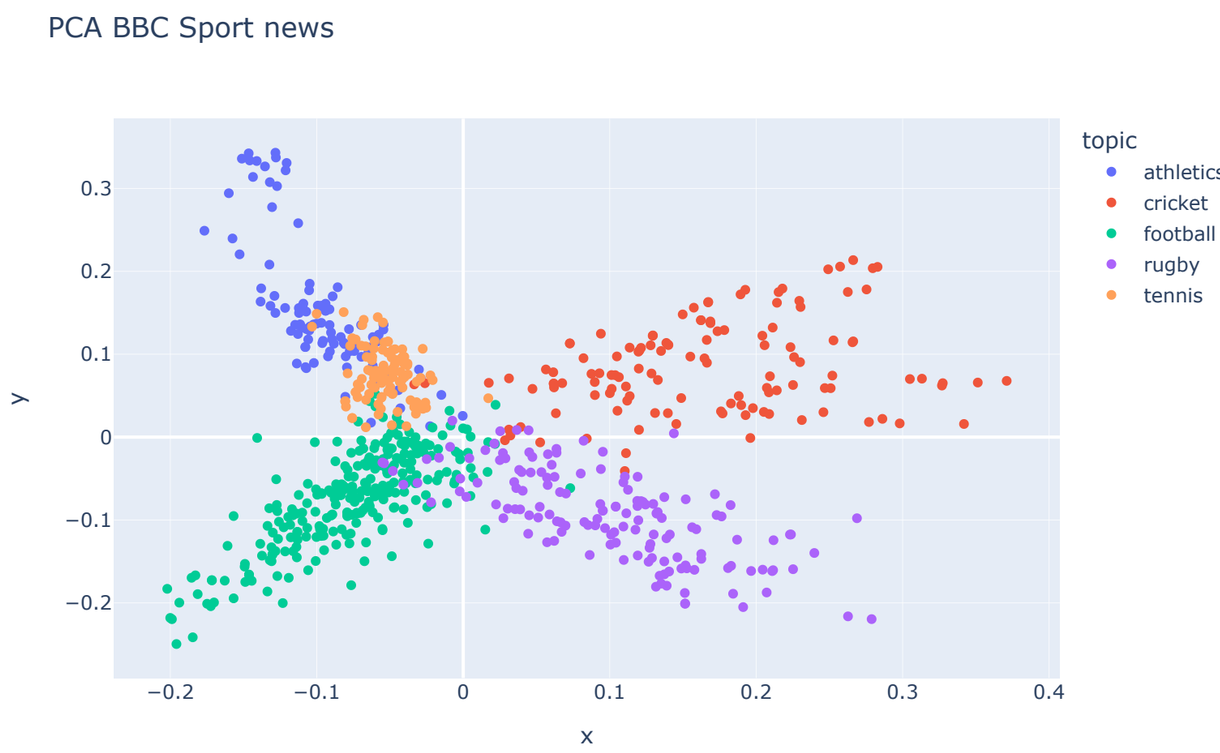
In [56]:
```
#Also, we can "visualize" the most common words for each topic with top_words
NUM_TOP_WORDS = 6
df.groupby('topic')['text'].apply(lambda x: hero.top_words(x)[:NUM_TOP_WORDS])
```

Out[56]:
```
topic
athletics  the    1731
           in      885
           to      882
           and     665
           a       613
           of      596
cricket    the    2246
           to     1309
           a      1045
           in     1007
           and     988
           of      834
football   the    4516
           to     2641
           a      2061
           and    1904
           in     1580
           of     1494
rugby      the    2770
           to     1393
           a      1210
           and    1129
           in     1093
           of      881
tennis     the    1527
           to      826
           in      706
           a       587
           and     573
           I       488
Name: text, dtype: int64
```

In [57]:
```
import texthero as hero
import pandas as pd

df = pd.read_csv(
    "https://github.com/jbesomi/texthero/raw/master/dataset/bbcsport.csv"
)
df['pca'] = (
    df['text']
    .pipe(hero.clean)
    .pipe(hero.tfidf)
    .pipe(hero.pca)
)

hero.scatterplot(df, col='pca', color='topic', title="PCA BBC Sport news")
```



Preprocessing

The texthero.preprocess module allow for efficient pre-processing of text-based Pandas Series and DataFrame.

1. clean(s[, pipeline])

Pre-process a text-based Pandas Series.

2. drop_no_content(s)

Drop all rows without content.

3. get_default_pipeline()

Return a list contaning all the methods used in the default cleaning pipeline.

4. has_content(s)

Return a Boolean Pandas Series indicating if the rows has content.

5. remove_angle_brackets(s)

Remove content within angle brackets <> and the angle brackets.

6. remove_brackets(s)

Remove content within brackets and the brackets itself.

7. remove_curly_brackets(s)

Remove content within curly brackets {} and the curly brackets.

8. remove_diacritics(input)

Remove all diacritics and accents.

9. remove_digits(input[, only_blocks])

Remove all digits and replace it with a single space.

10. remove_html_tags(s)

Remove html tags from the given Pandas Series.

11. remove_punctuation(input)

Replace all punctuation with a single space (" ").

12. remove_round_brackets(s)

Remove content within parentheses () and parentheses.

13. remove_square_brackets(s)

Remove content within square brackets [] and the square brackets.

14. remove_stopwords(input, stopwords, …[, …])

Remove all instances of words.

15. remove_urls(s)

Remove all urls from a given Pandas Series.

16. replace_urls(s, symbol)

Replace all urls with the given symbol.

17. remove_whitespace(input)

Remove any extra white spaces.

18. replace_punctuation(input, symbol)

Replace all punctuation with a given symbol.

19. replace_stopwords(input, symbol, stopwords, …)

Replace all instances of words with symbol.

20. tokenize(s)

Tokenize each row of the given Series.

More resources

1. https://texthero.org/docs/api-representation (https://texthero.org/docs/api-representation)
2. https://github.com/jbesomi/texthero (https://github.com/jbesomi/texthero)
3. https://pypi.org/project/texthero/ (https://pypi.org/project/texthero/)
4. https://github.com/573-pankaj/TextHero-Text-Preprocessing-in-NLP- (https://github.com/573-pankaj/TextHero-Text-Preprocessing-in-NLP-)
5. https://github.com/573-pankaj/NLP (https://github.com/573-pankaj/NLP)

In [ ]: