```
!pip install turicreate
```

```
from google.colab import drive
drive.mount('/content/gdrive')
```

```
Mounted at /content/gdrive
```

Document retrieval from Wikipedia data

```
import turicreate
```

Load some text data from Wikipedia

```
people = turicreate.SFrame('/content/gdrive/My Drive/Turicreate/Week 4/people_wiki.sframe')
```

```
people
```

URI	name	text
<http://dbpedia.org/resource/Digby_Morrell> ...	Digby Morrell	digby morrell born 10 october 1979 is a former ...
<http://dbpedia.org/resource/Alfred_J._Lewy> ...	Alfred J. Lewy	alfred j lewy aka sandy lewy graduated from ...
<http://dbpedia.org/resource/Harpdog_Brown> ...	Harpdog Brown	harpdog brown is a singer and harmonica player who ...
<http://dbpedia.org/resource/Franz_Rottensteiner> ...	Franz Rottensteiner	franz rottensteiner born in waidmannsfeld lower ...
<http://dbpedia.org/resource/G-Enka> ...	G-Enka	henry krvits born 30 december 1974 in tallinn ...
<http://dbpedia.org/resource/Sam_Henderson> ...	Sam Henderson	sam henderson born october 18 1969 is an ...
<http://dbpedia.org/resource/Aaron_LaCrate> ...	Aaron LaCrate	aaron lacrate is an american music producer ...
<http://dbpedia.org/resource/Trevor_Ferguson> ...	Trevor Ferguson	trevor ferguson aka john farrow born 11 november ...
<http://dbpedia.org/resource/Grant_Nelson> ...	Grant Nelson	grant nelson born 27 april 1971 in london ...
<http://dbpedia.org/resource/Cathy_Caruth> ...	Cathy Caruth	cathy caruth born 1955 is frank h t rhodes ...

[59071 rows x 3 columns]
Note: Only the head of the SFrame is printed.
You can use print_rows(num_rows=m, num_columns=n) to print more rows and columns.

Explore data

Taking a look at the entry for President Obama

```
obama = people[people['name'] == 'Barack Obama']
```

```
obama
```

URI	name	text
<http://dbpedia.org/resource/Barack_Obama> ...	Barack Obama	barack hussein obama ii brk husen bm born august ...

[? rows x 3 columns]
Note: Only the head of the SFrame is printed. This SFrame is lazily evaluated.
You can use sf.materialize() to force materialization.

```
obama['text']
```

```
dtype: str
Rows: ?
['barack hussein obama ii brk husen bm born august 4 1961 is the 44th and current president c
```

Explore the entry for actor George Clooney

```
clooney = people[people['name'] == 'George Clooney']
clooney['text']
```

```
dtype: str
Rows: ?
['george timothy clooney born may 6 1961 is an american actor writer producer director and ac
```

Word counts for Obama acticle

```
obama['word_count'] = turicreate.text_analytics.count_words(obama['text'])
```

```
obama
```

URI	name	text	word_count
<http://dbpedia.org/resource/Barack_Obama> ...	Barack Obama	barack hussein obama ii brk husen bm born august ...	{'normalize': 1.0, 'sought': 1.0, 'combat': ...

[1 rows x 4 columns]

```
print (obama['word_count'])
```

```
[{'normalize': 1.0, 'sought': 1.0, 'combat': 1.0, 'continued': 1.0, 'unconstitutional': 1.0,
```

Find most common words in Obama article

```
obama.stack('word_count',new_column_name=['word','count'])
```

URI	name	text	word	count
<http://dbpedia.org/resource/Barack_Obama> ...	Barack Obama	barack hussein obama ii brk husen bm born august ...	normalize	1.0
<http://dbpedia.org/resource/Barack_Obama> ...	Barack Obama	barack hussein obama ii brk husen bm born august ...	sought	1.0
<http://dbpedia.org/resource/Barack_Obama> ...	Barack Obama	barack hussein obama ii brk husen bm born august ...	combat	1.0
<http://dbpedia.org/resource/Barack_Obama> ...	Barack Obama	barack hussein obama ii brk husen bm born august ...	continued	1.0
<http://dbpedia.org/resource/Barack_Obama> ...	Barack Obama	barack hussein obama ii brk husen bm born august ...	unconstitutional	1.0
<http://dbpedia.org/resource/Barack_Obama> ...	Barack Obama	barack hussein obama ii brk husen bm born august ...	8	1.0

```
obama_word_count_table = obama[['word_count']].stack('word_count', new_column_name = ['word','count
```

```
obama_word_count_table
```

word	count
normalize	1.0
sought	1.0
combat	1.0
continued	1.0
unconstitutional	1.0
8	1.0
californias	1.0
1996	1.0
marriage	1.0
defense	1.0

[273 rows x 2 columns]
Note: Only the head of the SFrame is printed.
You can use print_rows(num_rows=m, num_columns=n) to print more rows and columns.

```
obama_word_count_table.sort('count',ascending=False)
```

word	count
the	40.0
in	30.0
and	21.0
of	18.0
to	14.0
his	11.0
obama	9.0
act	8.0
a	7.0
he	7.0

[273 rows x 2 columns]
Note: Only the head of the SFrame is printed.

Compute TF-IDF for the entire corpus of articles

```
people['word_count'] = turicreate.text_analytics.count_words(people['text'])
```

```
people
```

URI	name	text	word_count
<http://dbpedia.org/resource/Digby_Morrell> ...	Digby Morrell	digby morrell born 10 october 1979 is a former ...	{'melbourne': 1.0, 'parade': 1.0, ...
<http://dbpedia.org/resource/Alfred_J._Lewy> ...	Alfred J. Lewy	alfred j lewy aka sandy lewy graduated from ...	{'time': 1.0, 'each': 1.0, 'hour': 1.0, ...
<http://dbpedia.org/resource/Harpdog_Brown> ...	Harpdog Brown	harpdog brown is a singer and harmonica player who ...	{'society': 1.0, 'hamilton': 1.0, 'to': ...
<http://dbpedia.org/resource/Franz_Rottensteiner> ...	Franz Rottensteiner	franz rottensteiner born in waidmannsfeld lower ...	{'kurdlawitzpreis': 1.0, 'awarded': 1.0, '2004': ...
<http://dbpedia.org/resource/G-Enka> ...	G-Enka	henry krvits born 30 december 1974 in tallinn ...	{'curtis': 1.0, '2007': 1.0, 'cent': 1.0, ...
<http://dbpedia.org/resource/Sam_Henderson> ...	Sam Henderson	sam henderson born october 18 1969 is an ...	{'asses': 1.0, 'sic': 1.0, 'toilets': 1.0, ...
<http://dbpedia.org/resource/Aaron_LaCrate> ...	Aaron LaCrate	aaron lacrate is an american music producer ...	{'streamz': 1.0, 'including': 1.0, ...
<http://dbpedia.org/resou	Trevor	trevor ferguson aka	{'concordia': 1.0,

```
people['tfidf'] = turicreate.text_analytics.tf_idf(people['text'])
```

```
people
```

URI	name	text	word_count	tfidf
<http://dbpedia.org/resource/Digby_Morrell> ...	Digby Morrell	digby morrell born 10 october 1979 is a former ...	{'melbourne': 1.0, 'parade': 1.0, ...	{'melbourne': 3.8914310119380633, ...
<http://dbpedia.org/resource/Alfred_J._Lewy> ...	Alfred J. Lewy	alfred j lewy aka sandy lewy graduated from ...	{'time': 1.0, 'each': 1.0, 'hour': 1.0, ...	{'time': 1.3253342074200498, ...
<http://dbpedia.org/resource/Harpdog_Brown> ...	Harpdog Brown	harpdog brown is a singer and harmonica player who ...	{'society': 1.0, 'hamilton': 1.0, 'to': ...	{'society': 2.4448047262085693, ...
<http://dbpedia.org/resource/Franz_Rottensteiner> ...	Franz Rottensteiner	franz rottensteiner born in waidmannsfeld lower ...	{'kurdlawitzpreis': 1.0, 'awarded': 1.0, '2004': ...	
<http://dbpedia.org/resource/G-Enka> ...	G-Enka	henry krvits born 30 december 1974 in tallinn ...	{'curtis': 1.0, '2007': 1.0, 'cent': 1.0, ...	
<http://dbpedia.org/resource/Sam_Henderson> ...	Sam Henderson	sam henderson born october 18 1969 is an ...	{'asses': 1.0, 'sic': 1.0, 'toilets': 1.0, ...	
<http://dbpedia.org/resource/Aaron_LaCrate> ...	Aaron LaCrate	aaron lacrate is an american music producer ...	{'streamz': 1.0, 'including': 1.0, ...	
<http://dbpedia.org/resource/Trevor_Ferguson> ...	Trevor Ferguson	trevor ferguson aka john farrow born 11 november ...	{'concordia': 1.0, 'creative': 1.0, ...	
<http://dbpedia.org/resource/Grant_Nelson> ...	Grant Nelson	grant nelson born 27 april 1971 in london ...	{'heavies': 1.0, 'new': 1.0, 'brand': 1.0, ...	
<http://dbpedia.org/resource/Cathy_Caruth> ...	Cathy Caruth	cathy caruth born 1955 is frank h t rhodes ...	{'2002': 1.0, 'harvard': 1.0, 'twentieth': 1.0, ...	

Examine the TF-IDF for the Obama article

```
obama = people[people['name'] == 'Barack Obama']
obama[['tfidf']].stack('tfidf',new_column_name=['word','tfidf']).sort('tfidf',ascending=False)
```

word	tfidf
obama	43.2956530720749
act	27.67822262297991
iraq	17.747378587965535
control	14.887060845181308
law	14.722935761763422
ordered	14.533373950913514
military	13.115923778409415

Examine the TF-IDF for Clooney

```
clooney = people[people['name'] == 'George Clooney']
clooney[['tfidf']].stack('tfidf',new_column_name=['word','tfidf']).sort('tfidf',ascending=False)
```

word	tfidf
clooney	30.47679823695488
thriller	19.64459743254604
drama	13.544372218899177
comedydrama	12.973371437789858
er	12.782751078181208
actor	11.832160900443771
categoriesclooney	10.986495389225194
heslov	10.986495389225194
producingclooney	10.986495389225194
comedy	10.481205264908446

[239 rows x 2 columns]
Note: Only the head of the SFrame is printed.
You can use print_rows(num_rows=m, num_columns=n) to print more rows and columns.

Manually evaluate the distance between certain people's articles

```
clinton = people[people['name'] == 'Bill Clinton']
beckham = people[people['name'] == 'David Beckham']
```

Is Obama closer to Clinton or to Beckham?

```
turicreate.distances.cosine(obama['tfidf'][0],clinton['tfidf'][0])
```

```
0.8339854936884277
```

```
turicreate.distances.cosine(obama['tfidf'][0],beckham['tfidf'][0])
```

```
0.9791305844747478
```

Apply nearest neighbors for retrieval of Wikipedia articles

Build the NN model

```
knn_model = turicreate.nearest_neighbors.create(people,features=['tfidf'],label='name')
```

```
Starting brute force nearest neighbors model training.
Validating distance components.
Initializing model data.
Initializing distances.
Done.
```

Use model for retrieval... for example, who is closest to Obama?

```
knn_model.query(obama)
```

```
Starting pairwise querying.
+--------------+---------+-------------+--------------+
| Query points | # Pairs | % Complete. | Elapsed Time |
+--------------+---------+-------------+--------------+
| 0            | 1       | 0.00169288  | 17.894ms     |
| Done         |         | 100         | 457.249ms    |
+--------------+---------+-------------+--------------+
```

query_label	reference_label	distance	rank
0	Barack Obama	0.0	1
0	Joe Biden	0.7941176470588236	2
0	Joe Lieberman	0.7946859903381642	3
0	Kelly Ayotte	0.8119891008174387	4
0	Bill Clinton	0.8138528138528138	5

[5 rows x 4 columns]

Other examples of retrieval

```
swift = people[people['name'] == 'Taylor Swift']
```

```
knn_model.query(swift)
```

```
Starting pairwise querying.
+--------------+---------+--------------+--------------+
| Query points | # Pairs | % Complete.  | Elapsed Time |
+--------------+---------+--------------+--------------+
| 0            | 1       | 0.00169288   | 7.092ms      |
| Done         |         | 100          | 443.823ms    |
```

```
jolie = people[people['name'] == 'Angelina Jolie']
```

```
|       0       |      Taylor Swift      |           0.0           |    1    |
```

```
knn_model.query(jolie)
```

```
Starting pairwise querying.
+--------------+---------+--------------+--------------+
| Query points | # Pairs | % Complete.  | Elapsed Time |
+--------------+---------+--------------+--------------+
| 0            | 1       | 0.00169288   | 14.566ms     |
| Done         |         | 100          | 447.998ms    |
+--------------+---------+--------------+--------------+
```

query_label	reference_label	distance	rank
0	Angelina Jolie	0.0	1
0	Brad Pitt	0.7840236686390533	2
0	Julianne Moore	0.7958579881656804	3
0	Billy Bob Thornton	0.80306905370844	4
0	George Clooney	0.8046875	5

[5 rows x 4 columns]

```
arnold = people[people['name'] == 'Arnold Schwarzenegger']
```

```
knn_model.query(arnold)
```

```
Starting pairwise querying.
+--------------+---------+--------------+--------------+
| Query points | # Pairs | % Complete.  | Elapsed Time |
+--------------+---------+--------------+--------------+
| 0            | 1       | 0.00169288   | 10.374ms     |
| Done         |         | 100          | 466.549ms    |
+--------------+---------+--------------+--------------+
```

query_label	reference_label	distance	rank
0	Arnold Schwarzenegger	0.0	1
0	Jesse Ventura	0.8189189189189189	2
0	John Kitzhaber	0.8246153846153846	3
0	Lincoln Chafee	0.8338762214983714	4
0	Anthony Foxx	0.8339100346020761	5

[5 rows x 4 columns]

✓ 0초 오후 5:50에 완료됨