We'll be covering text classification and regression methods over the next month; in preparation for this topic, your assignment is to gather labeled data to use for your analysis.

- Find at least 300 documents for some topic that interests you, along with a single binary label for each document. Aim high if you can; the more data in your collection, the better your classification models will tend to perform on it.
- Split your data into three non-overlapping files (train.tsv, dev.tsv and test.tsv), with train.tsv containing 80% of the documents, dev.tsv 10% and test.tsv 10%.
- All of the data must be in a common format; we'll use a tab-separated format with the label in the first column and the full text in the second column. Replace all newlines in the text with _NEWLINE_ and tab characters with _TAB_.

See data/text_classification_sample/ for an example. Execute this Jupyter notebook to verify that your format is correct.

Your choice of documents and labels is completely up to you (except for any data already used in class in the data/ folder). Possible sources of data:

- Project Gutenberg. Metadata is available at this Github repo along with URLs for the texts. Labels here can be author, subject, author gender etc.
- Crawl news articles from different domains (e.g,. CNN, FoxNews); the label for each article is the domain.
- Movie summary data. Labels here can be any categorical metadata aspect (genre, release date); note real-valued metadata (like box office, runtime) can be binarized by selecting some threshold.
- Download your own tweets. Labels here can be any categorical metadata included in the tweet, or labels you add by hand (e.g., sarcasm)

In [1]:
```
import sys
from collections import Counter
```

In [2]:
```
def test(directory):
    for split in ["train", "dev", "test"]:
        filename="%s/%s.tsv" % (directory, split)
        with open(filename,encoding='utf-8') as file:
            labelCounts=Counter()
            zeroLength=0
            total=0
            for line in file:
                cols=line.rstrip().split("\t")
                label=cols[0]
                text=cols[1]
                if len(text) == 0:
                    zeroLength+=1
                total+=1
```

```
            labelCounts[label]+=1

        print ("File: %s, Total docs: %s, Total zero length: %s" % (filename, total
        for label in sorted(labelCounts):
            print ("\t%s %s" % (label, labelCounts[label]))
        print()
```

Q1: Describe your data. What is the source of the documents, and what do the labels mean?

I noticed that some TED talks are more persuasive— they change your day and maybe your life. Some fall flat and are completely forgettable. Can we use the tools of natural language processing and statistical models to understand why some talks work and to see how persuaders persuade?

In my original collected metadata, held a variable called 'persuasive' that paried a value of the number of TED.com users who had voted a particular talk persuasive using TED.com's ratings tool. To normalize the ratings and account for the fact that talks have not been viewed the same number of times (i.e., some talks have been posted for months and others for years), I divided the count of persuasive votes by the number of times the talk had been viewed to create the 'norm_persuasive' variable. This variable is persuasive votes per view of a talk.

- To convert this normalized value to 0, 1 categoricle lable, I picked the median 91 to be the seperation line, values below this number will accounted as 0, not persuasive, and 1 as persuasive.
- The size of the whole dataset has 2385 data entries, after the test, triam, dev split, we have 1908 for trianing data, 238 for test dato and 239 for dev data.

Q2: Change the directionary name below to the directory containing your data and execute the `test()` function above to verify the data is in the correct format:

In [4]:
```
import csv
import pandas as pd
```

In [37]:
```
#import the csv file
df = pd.read_csv("../data/ted/all.csv", header= None, index_col = False )
df.rename(columns={0: 'Persuasive', 1: 'data'}, inplace=True)
```

In [38]:
```
#checking the dimension and labels
df
```

Out[38]:

	Persuasive	data
0	1	I have a very difficult task. I'm a spectrosco...
1	0	I run a design studio in New York. Every seven...
2	1	How do you feed a city? It's one of the great ...
3	1	What we're really here to talk about is the "h...
4	1	I'm a storyteller. And I would like to tell yo...

	Persuasive	data
...	...	...
2380	1	Who are we? That is the big question. And esse...
2381	0	So I'm going to talk today about collecting st...
2382	0	To be new at TED "it's like being the last hi...
2383	1	The Internet, the Web as we know it, the kind ...
2384	1	I got my first computer when I was a teenager ...

2385 rows × 2 columns

In [39]:
```
#checking if there's new line in data
"\n" in df['data']
```

Out[39]: False

In [40]:
```
#checking if there's tab in data
"\t" in df['data']
```

Out[40]: False

In [48]:
```
#split train and test_dev set
from sklearn.model_selection import train_test_split
train, test_dev = train_test_split(df, test_size=0.2, random_state=42, shuffle=True)
```

In [49]:
```
#split test and dev set
test, dev = train_test_split(test_dev, test_size=0.5, random_state=42, shuffle=True)
```

In [51]:
```
train
```

Out[51]:

	Persuasive	data
275	1	So, I'll start with this: a couple years ago, ...
2107	0	I have a studio in Berlin "let me cue on here ...
1406	0	A few years ago, with my colleague, Emmanuelle...
360	0	Good afternoon, everybody. I've got something ...
1711	0	I want to introduce you to some very wise kids...
...	...	...
1638	0	I'm here today to talk to you about a very pow...
1095	1	There's a man by the name of Captain William S...
1130	0	Nicholas Negroponte: Can we switch to the v...
1294	1	When I was a kid, the disaster we worried abou...

	Persuasive	data
860	1	In this talk today, I want to present a differ...

1908 rows × 2 columns

In [52]:
```
test
```

Out[52]:

	Persuasive	data
486	1	I'd like to apologize, first of all, to all of...
443	1	I want to say that really and truly, after the...
575	1	The electricity powering the lights in this th...
1793	0	I'm a journalist, and I'm an immigrant. And th...
1728	0	It was just an ordinary Saturday. My dad was o...
...	...	...
1651	1	I am so excited to be here. Everything in Amer...
1041	1	I'm going to talk about hackers. And the image...
1882	1	As you've probably noticed, in recent years, ...
1714	1	When I was a kid, I was obsessed with the Guin...
1090	1	So today's top chef class is in how to rob a b...

238 rows × 2 columns

In [53]:
```
dev
```

Out[53]:

	Persuasive	data
283	1	I would like to tell you all that you are all ...
874	1	There's something that I'd like you to see. R...
134	0	We invent. My company invents all kinds of new...
1288	0	I'm a potter, which seems like a fairly humble...
1912	0	What are you doing on this stage in front of ...
...	...	...
1874	0	I've been doing some thinking. I'm going to k...
2352	1	So, can we dare to be optimistic? Well, the th...
1127	1	When we think about prejudice and bias, we ten...
508	0	I study how the brain processes information. T...
1864	0	You're watching the life cycle of a Streptomy...

239 rows × 2 columns

In [59]:
```
#save as .tsv file
dev.to_csv('dev.tsv', sep = '\t', index=False, header=False)
```

In [60]:
```
test.to_csv('test.tsv', sep = '\t', index=False, header=False)
```

In [61]:
```
train.to_csv('train.tsv', sep = '\t', index=False, header=False)
```

In [3]:
```
#setting directory and run test
directory="../data/ted"
```

In [4]:
```
test(directory)
```

```
File: ../data/ted/train.tsv, Total docs: 1908, Total zero length: 0
       0 979
       1 929

File: ../data/ted/dev.tsv, Total docs: 239, Total zero length: 0
       0 112
       1 127

File: ../data/ted/test.tsv, Total docs: 238, Total zero length: 0
       0 104
       1 134
```