Data Cleaning

Introduction

This notebook goes through a necessary step of any data science project - data cleaning. Data cleaning is a time consuming and unenjoyable task, yet it's a very important one. Keep in mind, "garbage in, garbage out". Feeding dirty data into a model will give us results that are meaningless.

Specifically, we'll be walking through:

1. **Getting the data* - *in this case, we'll be scraping data from a website
2. **Cleaning the data* - *we will walk through popular text pre-processing techniques
3. **Organizing the data* - *we will organize the cleaned data into a way that is easy to input into other algorithms

The output of this notebook will be clean, organized data in two standard text formats:

1. **Corpus** - a collection of text
2. **Document-Term Matrix** - word counts in matrix format

Problem Statement

As a reminder, our goal is to look at transcripts of various comedians and note their similarities and differences. Specifically, I'd like to know if Ali Wong's comedy style is different than other comedians, since she's the comedian that got me interested in stand up comedy.

Getting The Data

Luckily, there are wonderful people online that keep track of stand up routine transcripts. Scraps From The Loft (http://scrapsfromtheloft.com) makes them available for non-profit and educational purposes.

To decide which comedians to look into, I went on IMDB and looked specifically at comedy specials that were released in the past 5 years. To narrow it down further, I looked only at those with greater than a 7.5/10 rating and more than 2000 votes. If a comedian had multiple specials that fit those requirements, I would pick the most highly rated one. I ended up with a dozen comedy specials.

In [2]:
```
# Web scraping, pickle imports
import requests
from bs4 import BeautifulSoup
import pickle

# Scrapes transcript data from scrapsfromtheloft.com
def url_to_transcript(url):
    '''Returns transcript data specifically from scrapsfromtheloft.com.'''
    page = requests.get(url).text
    soup = BeautifulSoup(page, "lxml")
    text = [p.text for p in soup.find(class_="post-content").find_all('p')]
    print(url)
    return text

# URLs of transcripts in scope
urls = ['http://scrapsfromtheloft.com/2017/05/06/louis-ck-oh-my-god-full-transcript/',
        'http://scrapsfromtheloft.com/2017/04/11/dave-chappelle-age-spin-2017-full-transcript/',
        'http://scrapsfromtheloft.com/2018/03/15/ricky-gervais-humanity-transcript/',
        'http://scrapsfromtheloft.com/2017/08/07/bo-burnham-2013-full-transcript/',
        'http://scrapsfromtheloft.com/2017/05/24/bill-burr-im-sorry-feel-way-2014-full-transcript/',
        'http://scrapsfromtheloft.com/2017/04/21/jim-jefferies-bare-2014-full-transcript/',
        'http://scrapsfromtheloft.com/2017/08/02/john-mulaney-comeback-kid-2015-full-transcript/',
        'http://scrapsfromtheloft.com/2017/10/21/hasan-minhaj-homecoming-king-2017-full-transcript/',
        'http://scrapsfromtheloft.com/2017/09/19/ali-wong-baby-cobra-2016-full-transcript/',
        'http://scrapsfromtheloft.com/2017/08/03/anthony-jeselnik-thoughts-prayers-2015-full-transcript/',
        'http://scrapsfromtheloft.com/2018/03/03/mike-birbiglia-my-girlfriends-boyfriend-2013-full-transcript/',
        'http://scrapsfromtheloft.com/2017/08/19/joe-rogan-triggered-2016-full-transcript/']

# Comedian names
comedians = ['louis', 'dave', 'ricky', 'bo', 'bill', 'jim', 'john', 'hasan', 'ali', 'anthony', 'mike', 'joe']
```

In [ ]:
```
# # Actually request transcripts (takes a few minutes to run)
# transcripts = [url_to_transcript(u) for u in urls]
```

In [ ]:
```
# # Pickle files for later use

# # Make a new directory to hold the text files
# !mkdir transcripts

# for i, c in enumerate(comedians):
#     with open("transcripts/" + c + ".txt", "wb") as file:
#         pickle.dump(transcripts[i], file)
```

In [3]:
```
# Load pickled files
data = {}
for i, c in enumerate(comedians):
    with open("transcripts/" + c + ".txt", "rb") as file:
        data[c] = pickle.load(file)
```

In [4]:
```
# Double check to make sure data has been loaded properly
data.keys()
```

Out[4]:
```
dict_keys(['louis', 'dave', 'ricky', 'bo', 'bill', 'jim', 'john', 'hasan', 'ali', 'anthony', 'mike', 'joe'])
```

In [5]:
```
# More checks
data['louis'][:2]
```

Out[5]:
```
['Intro\nFade the music out. Let’s roll. Hold there. Lights. Do the lights. Thank you. Thank you very much. I appreciate that. I don’t necessarily ag
ree with you, but I appreciate very much. Well, this is a nice place. This is easily the nicest place For many miles in every direction. That’s how y
ou compliment a building And shit on a town with one sentence. It is odd around here, as I was driving here. There doesn’t seem to be any difference
Between the sidewalk and the street for pedestrians here. People just kind of walk in the middle of the road. I love traveling And seeing all the dif
ferent parts of the country. I live in New York. I live in a— There’s no value to your doing that at all.',
 '“The Old Lady And The Dog”\nI live— I live in New York. I always— Like, there’s this old lady in my neighborhood, And she’s always walking her dog.
She’s always just— she’s very old. She just stands there just being old, And the dog just fights gravity every day, just— The two of them, it’s reall
y— The dog’s got a cloudy eye, and she’s got a cloudy eye, And they just stand there looking at the street In two dimensions together, and— And she’s
always wearing, like, this old sweater dress. I guess it was a sweater when she was, like, 5’10”, But now it’s just, like, this sweater And her legs
are— her legs are a nightmare. They’re just white with green streaks and bones sticking out. Her legs are awful. I saw a guy with no legs wheeling b
y, And he was like, “yecch, no thank you. “I do not want those. “I’d rather just have air down here like I have Than to look down at that shit.” I se
e these two all the time, and I always look at them, And I always think, “god, I hope she dies first.” I do. I hope she dies first, for her sake, Bec
ause I don’t want her to lose the dog. I don’t think she’ll be able to handle it. If she dies— If the old lady dies first, I’m not worried about the
dog Because the dog doesn’t even know about the old lady. This dog is aware of three inches around his head. He’s living in two-second increments. Th
e second he’s in and the one he just left Is all he knows about, But if he dies, this lady, she’s gonna be destroyed Because this dog is all she has,
And I know he’s all she has because she has him. There’s no— If she had one person in her life, She would not keep this piece of shit little dog. Eve
n if just some young woman in her building one morning Were to say, “good morning, gladys,” She’d be like, “good,” And just flush him down the toile
t, just— Poom! Poom! The dog just keeps bumping on the drain. Poom! “” she gives up. Ends up just shitting on her dog for the rest of her life. P-p-
p! Poom!']
```

Cleaning The Data

When dealing with numerical data, data cleaning often involves removing null values and duplicate data, dealing with outliers, etc. With text data, there are some common data cleaning techniques, which are also known as text pre-processing techniques.

With text data, this cleaning process can go on forever. There's always an exception to every cleaning step. So, we're going to follow the MVP (minimum viable product) approach - start simple and iterate. Here are a bunch of things you can do to clean your data. We're going to execute just the common cleaning steps here and the rest can be done at a later point to improve our results.

Common data cleaning steps on all text:

- Make text all lower case
- Remove punctuation
- Remove numerical values
- Remove common non-sensical text (/n)
- Tokenize text
- Remove stop words

More data cleaning steps after tokenization:

- Stemming / lemmatization
- Parts of speech tagging
- Create bi-grams or tri-grams
- Deal with typos
- And more...

In [6]:
```
# Let's take a look at our data again
next(iter(data.keys()))
```

Out[6]:
```
'louis'
```

In [7]:
```
# Notice that our dictionary is currently in key: comedian, value: list of text format
next(iter(data.values()))
```

Out[7]:
```
['Intro\nFade the music out. Let’s roll. Hold there. Lights. Do the lights. Thank you. Thank you very much. I appreciate that. I don’t necessarily
agree with you, but I appreciate very much. Well, this is a nice place. This is easily the nicest place For many miles in every direction. That’s h
ow you compliment a building And shit on a town with one sentence. It is odd around here, as I was driving here. There doesn’t seem to be any diffe
rence Between the sidewalk and the street for pedestrians here. People just kind of walk in the middle of the road. I love traveling And seeing all
the different parts of the country. I live in New York. I live in a— There’s no value to your doing that at all.',
 '“The Old Lady And The Dog”\nI live— I live in New York. I always— Like, there’s this old lady in my neighborhood, And she’s always walking her do
g. She’s always just— she’s very old. She just stands there just being old, And the dog just fights gravity every day, just— The two of them, it’s
really— The dog’s got a cloudy eye, and she’s got a cloudy eye, And they just stand there looking at the street In two dimensions together, and— An
d she’s always wearing, like, this old sweater dress. I guess it was a sweater when she was, like, 5’10”, But now it’s just, like, this sweater And
her legs are— her legs are a nightmare. They’re just white with green streaks and bones sticking out. Her legs are awful. I saw a guy with no legs
wheeling by, And he was like, “yecch, no thank you. “I do not want those. “I’d rather just have air down here like I have Than to look down at that
shit.” I see these two all the time, and I always look at them, And I always think, “god, I hope she dies first.” I do. I hope she dies first, for
her sake, Because I don’t want her to lose the dog. I don’t think she’ll be able to handle it. If she dies— If the old lady dies first, I’m not wor
ried about the dog Because the dog doesn’t even know about the old lady. This dog is aware of three inches around his head. He’s living in two-seco
nd increments. The second he’s in and the one he just left Is all he knows about, But if he dies, this lady, she’s gonna be destroyed Because this
dog is all she has, And I know he’s all she has because she has him. There’s no— If she had one person in her life, She would not keep this piece o
f shit little dog. Even if just some young woman in her building one morning Were to say, “good morning, gladys,” She’d be like, “good,” And just f
lush him down the toilet, just— Poom! Poom! The dog just keeps bumping on the drain. Poom! “” she gives up. Ends up just shitting on her dog for th
e rest of her life. P-p-p! Poom!',
 '“My Daughter Likes Fish”\nYou ever flush a pet down the toilet? I had to flush my daughter’s fish down the toilet. I came home, the fish was dea
```

In [8]:
```
# We are going to change this to key: comedian, value: string format
def combine_text(list_of_text):
    '''Takes a list of text and combines them into one large chunk of text.'''
    combined_text = ' '.join(list_of_text)
    return combined_text
```

In [9]:
```
# Combine it!
data_combined = {key: [combine_text(value)] for (key, value) in data.items()}
```

In [11]:
```
# We can either keep it in dictionary format or put it into a pandas dataframe
import pandas as pd
pd.set_option('max_colwidth',150)

data_df = pd.DataFrame.from_dict(data_combined).transpose()
data_df.columns = ['transcript']
data_df = data_df.sort_index()
data_df
```

Out[11]:

	transcript
ali	Ladies and gentlemen, please welcome to the stage: Ali Wong! Hi. Hello! Welcome! Thank you! Thank you for coming. Hello! Hello. We are gonna have ...
anthony	Thank you. Thank you. Thank you, San Francisco. Thank you so much. So good to be here. People were surprised when I told 'em I was gonna tape my s...
bill	[cheers and applause] All right, thank you! Thank you very much! Thank you. Thank you. Thank you. How are you? What's going on? Thank you. It's a ...
bo	Bo What? Old MacDonald had a farm E I E I O And on that farm he had a pig E I E I O Here a snort There a Old MacDonald had a farm E I E I O [Appla...
dave	This is Dave. He tells dirty jokes for a living. That stare is where most of his hard work happens. It signifies a profound train of thought, the ...
hasan	[theme music: orchestral hip-hop] [crowd roars] What's up? Davis, what's up? I'm home. I had to bring it back here. Netflix said, "Where do you wa...
jim	[Car horn honks] [Audience cheering] [Announcer] Ladies and gentlemen, please welcome to the stage Mr. Jim Jefferies! [Upbeat music playing] Hello...
joe	[rock music playing] [audience cheering] [announcer] Ladies and gentlemen, welcome Joe Rogan. [audience cheering and applauding] What the fuck is ...
john	All right, Petunia. Wish me luck out there. You will die on August 7th, 2037. That's pretty good. All right. Hello. Hello, Chicago. Nice to see yo...
louis	Intro\nFade the music out. Let's roll. Hold there. Lights. Do the lights. Thank you. Thank you very much. I appreciate that. I don't necessarily a...
mike	Wow. Hey, thank you. Thanks. Thank you, guys. Hey, Seattle. Nice to see you. Look at this. Look at us. We're here. This is crazy. It's insane. So ...
ricky	Hello. Hello! How you doing? Great. Thank you. Wow. Calm down. Shut the fuck up. Thank you. What a lovely welcome. I'm gonna try my hardest tonigh...

In [12]:
```
# Let's take a look at the transcript for Ali Wong
data_df.transcript.loc['ali']
```

Out[12]:
```
"Ladies and gentlemen, please welcome to the stage: Ali Wong! Hi. Hello! Welcome! Thank you! Thank you for coming. Hello! Hello. We are gonna have
to get this shit over with, 'cause I have to pee in, like, ten minutes. But thank you, everybody, so much for coming. Um… It's a very exciting day
for me. It's been a very exciting year for me. I turned 33 this year. Yes! Thank you, five people. I appreciate that. Uh, I can tell that I'm getti
ng older, because, now, when I see an 18-year-old girl, my automatic thought… is "Fuck you." "Fuck you. I don't even know you, but fuck you!" 'Caus
e I'm straight up jealous. I'm jealous, first and foremost, of their metabolism. Because 18-year-old girls, they could just eat like shit, and then
they take a shit and have a six-pack, right? They got that-that beautiful inner thigh clearance where they put their feet together and there's that
huge gap here with the light of potential just radiating through.\nAnd then, when they go to sleep, they just go to sleep. Right? They don't have i
nsomnia yet. They don't know what it's like to have to take a Ambien or download a Meditation Oasis podcast to calm the chatter of regret and resen
tment towards your family just cluttering your mind. They have their whole lives ahead of them. They don't have HPV yet. They just go to sleep in p
eace at night. Everybody has HPV, OK? Everybody has it. It's OK. Come out already. Everybody has it. If you don't have it yet, you go and get it. Y
ou go and get it. It's coming. You don't have HPV yet, you're a fucking loser, all right? That's what that says about you. A lot of men don't know
that they have HPV, because it's undetectable in men. It's really fucked up. HPV is a ghost that lives inside men's bodies and says, "Boo!" in wome
n's bodies. My doctor told me that I have one of two strains of HPV. Either I have the kind that's gonna turn into cervical cancer… …or I have the
kind where my body will heal itself. Very helpful, this doctor, right? So, basically, either I'm gonna die… or you're in the presence of Wolverine,
bitches. We'll find out. Um, I can also tell that I'm getting older, because my Kindle is turning into a self-help library. I'm not interested in b
ooks like Fifty Shades of Grey, OK? I'm interested in The Life-Changing Magic of Tidying Up. Yes. Yes, that's right, how to declutter my home to ac
hieve inner peace and my optimum level of success. That's what your 30s is all about. How can I turn this shit around? I'm a horrible person, I'm n
ot happy with where I am, how can I turn this shit around? Help me, Tony Robbins, help me!\nI have a hoarding problem, which I'm hoping is the cent
er of all of my other problems. I'm hoping that if the hoarding goes away, the HPV will also disappear. I have a hoarding problem because my mom is
from a third world country and she taught me that you can never throw away anything, because you never know when a dictator's gonna overtake the co
```

In [13]:

```
# Apply a first round of text cleaning techniques
import re
import string

def clean_text_round1(text):
    '''Make text lowercase, remove text in square brackets, remove punctuation and remove words containing numbers.'''
    text = text.lower()
    text = re.sub('\[.*?\]', '', text)
    text = re.sub('[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub('\w*\d\w*', '', text)
    return text

round1 = lambda x: clean_text_round1(x)
```

In [14]:

```
# Let's take a look at the updated text
data_clean = pd.DataFrame(data_df.transcript.apply(round1))
data_clean
```

Out[14]:

	transcript
ali	ladies and gentlemen please welcome to the stage ali wong hi hello welcome thank you thank you for coming hello hello we are gonna have to get thi...
anthony	thank you thank you thank you san francisco thank you so much so good to be here people were surprised when i told 'em i was gonna tape my special...
bill	all right thank you thank you very much thank you thank you thank you how are you what's going on thank you it's a pleasure to be here in the gre...
bo	bo what old macdonald had a farm e i e i o and on that farm he had a pig e i e i o here a snort there a old macdonald had a farm e i e i o this i...
dave	this is dave he tells dirty jokes for a living that stare is where most of his hard work happens it signifies a profound train of thought the alch...
hasan	what's up davis what's up i'm home i had to bring it back here netflix said "where do you want to do the special la chicago new york" i was like...
jim	ladies and gentlemen please welcome to the stage mr jim jefferies hello sit down sit down sit down sit down sit down thank you boston i appre...
joe	ladies and gentlemen welcome joe rogan what the fuck is going on san francisco thanks for coming i appreciate it god damn put your phone down ...
john	all right petunia wish me luck out there you will die on august that's pretty good all right hello hello chicago nice to see you again thank you...
louis	intro\nfade the music out let's roll hold there lights do the lights thank you thank you very much i appreciate that i don't necessarily agree wit...
mike	wow hey thank you thanks thank you guys hey seattle nice to see you look at this look at us we're here this is crazy it's insane so about five yea...
ricky	hello hello how you doing great thank you wow calm down shut the fuck up thank you what a lovely welcome i'm gonna try my hardest tonight you're t...

In [15]:

```
# Apply a second round of cleaning
def clean_text_round2(text):
    '''Get rid of some additional punctuation and non-sensical text that was missed the first time around.'''
    text = re.sub('[‘’“”…]', '', text)
    text = re.sub('\n', '', text)
    return text

round2 = lambda x: clean_text_round2(x)
```

In [16]:
```
# Let's take a look at the updated text
data_clean = pd.DataFrame(data_clean.transcript.apply(round2))
data_clean
```

Out[16]:

	transcript
ali	ladies and gentlemen please welcome to the stage ali wong hi hello welcome thank you thank you for coming hello hello we are gonna have to get thi...
anthony	thank you thank you thank you san francisco thank you so much so good to be here people were surprised when i told em i was gonna tape my special ...
bill	all right thank you thank you very much thank you thank you thank you how are you whats going on thank you its a pleasure to be here in the great...
bo	bo what old macdonald had a farm e i e i o and on that farm he had a pig e i e i o here a snort there a old macdonald had a farm e i e i o this i...
dave	this is dave he tells dirty jokes for a living that stare is where most of his hard work happens it signifies a profound train of thought the alch...
hasan	whats up davis whats up im home i had to bring it back here netflix said where do you want to do the special la chicago new york i was like nah ...
jim	ladies and gentlemen please welcome to the stage mr jim jefferies hello sit down sit down sit down sit down sit down thank you boston i appre...
joe	ladies and gentlemen welcome joe rogan what the fuck is going on san francisco thanks for coming i appreciate it god damn put your phone down ...
john	all right petunia wish me luck out there you will die on august thats pretty good all right hello hello chicago nice to see you again thank you ...
louis	introfade the music out lets roll hold there lights do the lights thank you thank you very much i appreciate that i dont necessarily agree with yo...
mike	wow hey thank you thanks thank you guys hey seattle nice to see you look at this look at us were here this is crazy its insane so about five years...
ricky	hello hello how you doing great thank you wow calm down shut the fuck up thank you what a lovely welcome im gonna try my hardest tonight youre thi...

NOTE: This data cleaning aka text pre-processing step could go on for a while, but we are going to stop for now. After going through some analysis techniques, if you see that the results don't make sense or could be improved, you can come back and make more edits such as:

- Mark 'cheering' and 'cheer' as the same word (stemming / lemmatization)
- Combine 'thank you' into one term (bi-grams)
- And a lot more...

Organizing The Data

I mentioned earlier that the output of this notebook will be clean, organized data in two standard text formats:

1. **Corpus* - *a collection of text
2. **Document-Term Matrix* - *word counts in matrix format

Corpus

We already created a corpus in an earlier step. The definition of a corpus is a collection of texts, and they are all put together neatly in a pandas dataframe here.

In [17]:
```
# Let's take a look at our dataframe
data_df
```

Out[17]:

	transcript
ali	Ladies and gentlemen, please welcome to the stage: Ali Wong! Hi. Hello! Welcome! Thank you! Thank you for coming. Hello! Hello. We are gonna have ...
anthony	Thank you. Thank you. Thank you, San Francisco. Thank you so much. So good to be here. People were surprised when I told 'em I was gonna tape my s...
bill	[cheers and applause] All right, thank you! Thank you very much! Thank you. Thank you. Thank you. How are you? What's going on? Thank you. It's a ...
bo	Bo What? Old MacDonald had a farm E I E I O And on that farm he had a pig E I E I O Here a snort There a Old MacDonald had a farm E I E I O [Appla...
dave	This is Dave. He tells dirty jokes for a living. That stare is where most of his hard work happens. It signifies a profound train of thought, the ...
hasan	[theme music: orchestral hip-hop] [crowd roars] What's up? Davis, what's up? I'm home. I had to bring it back here. Netflix said, "Where do you wa...
jim	[Car horn honks] [Audience cheering] [Announcer] Ladies and gentlemen, please welcome to the stage Mr. Jim Jefferies! [Upbeat music playing] Hello...
joe	[rock music playing] [audience cheering] [announcer] Ladies and gentlemen, welcome Joe Rogan. [audience cheering and applauding] What the fuck is ...
john	All right, Petunia. Wish me luck out there. You will die on August 7th, 2037. That's pretty good. All right. Hello. Hello, Chicago. Nice to see yo...
louis	Intro\nFade the music out. Let's roll. Hold there. Lights. Do the lights. Thank you. Thank you very much. I appreciate that. I don't necessarily a...
mike	Wow. Hey, thank you. Thanks. Thank you, guys. Hey, Seattle. Nice to see you. Look at this. Look at us. We're here. This is crazy. It's insane. So ...
ricky	Hello. Hello! How you doing? Great. Thank you. Wow. Calm down. Shut the fuck up. Thank you. What a lovely welcome. I'm gonna try my hardest tonigh...

In [18]:
```
# Let's add the comedians' full names as well
full_names = ['Ali Wong', 'Anthony Jeselnik', 'Bill Burr', 'Bo Burnham', 'Dave Chappelle', 'Hasan Minhaj',
              'Jim Jefferies', 'Joe Rogan', 'John Mulaney', 'Louis C.K.', 'Mike Birbiglia', 'Ricky Gervais']

data_df['full_name'] = full_names
data_df
```

Out[18]:

	transcript	full_name
ali	Ladies and gentlemen, please welcome to the stage: Ali Wong! Hi. Hello! Welcome! Thank you! Thank you for coming. Hello! Hello. We are gonna have ...	Ali Wong
anthony	Thank you. Thank you. Thank you, San Francisco. Thank you so much. So good to be here. People were surprised when I told 'em I was gonna tape my s...	Anthony Jeselnik
bill	[cheers and applause] All right, thank you! Thank you very much! Thank you. Thank you. Thank you. How are you? What's going on? Thank you. It's a ...	Bill Burr
bo	Bo What? Old MacDonald had a farm E I E I O And on that farm he had a pig E I E I O Here a snort There a Old MacDonald had a farm E I E I O [Appla...	Bo Burnham
dave	This is Dave. He tells dirty jokes for a living. That stare is where most of his hard work happens. It signifies a profound train of thought, the ...	Dave Chappelle
hasan	[theme music: orchestral hip-hop] [crowd roars] What's up? Davis, what's up? I'm home. I had to bring it back here. Netflix said, "Where do you wa...	Hasan Minhaj
jim	[Car horn honks] [Audience cheering] [Announcer] Ladies and gentlemen, please welcome to the stage Mr. Jim Jefferies! [Upbeat music playing] Hello...	Jim Jefferies
joe	[rock music playing] [audience cheering] [announcer] Ladies and gentlemen, welcome Joe Rogan. [audience cheering and applauding] What the fuck is ...	Joe Rogan
john	All right, Petunia. Wish me luck out there. You will die on August 7th, 2037. That's pretty good. All right. Hello. Hello, Chicago. Nice to see yo...	John Mulaney
louis	Intro\nFade the music out. Let's roll. Hold there. Lights. Do the lights. Thank you. Thank you very much. I appreciate that. I don't necessarily a...	Louis C.K.
mike	Wow. Hey, thank you. Thanks. Thank you, guys. Hey, Seattle. Nice to see you. Look at this. Look at us. We're here. This is crazy. It's insane. So ...	Mike Birbiglia
ricky	Hello. Hello! How you doing? Great. Thank you. Wow. Calm down. Shut the fuck up. Thank you. What a lovely welcome. I'm gonna try my hardest tonigh...	Ricky Gervais

In [19]:
```
# Let's pickle it for later use
data_df.to_pickle("corpus.pkl")
```

Document-Term Matrix

For many of the techniques we'll be using in future notebooks, the text must be tokenized, meaning broken down into smaller pieces. The most common tokenization technique is to break down text into words. We can do this using scikit-learn's CountVectorizer, where every row will represent a different document and every column will represent a different word.

In addition, with CountVectorizer, we can remove stop words. Stop words are common words that add no additional meaning to text such as 'a', 'the', etc.

In [22]:
```
# We are going to create a document-term matrix using CountVectorizer, and exclude common English stop words
from sklearn.feature_extraction.text import CountVectorizer

cv = CountVectorizer(stop_words='english')
data_cv = cv.fit_transform(data_clean.transcript)
data_dtm = pd.DataFrame(data_cv.toarray(), columns=cv.get_feature_names())
data_dtm.index = data_clean.index
data_dtm
```

Out[22]:

	aaaaah	aaaaahhhhhhh	aaaaauuugghhhhhh	aaaahhhhh	aaah	aah	abc	abcs	ability	abject	...	zee	zen	zeppelin	zero	zillion	zombie	zombies	zoning	zoo	éclair
ali	0	0	0	0	0	0	1	0	0	0	...	0	0	0	0	0	1	0	0	0	0
anthony	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
bill	1	0	0	0	0	0	0	1	0	0	...	0	0	0	1	1	1	1	1	0	0
bo	0	1	1	1	0	0	0	0	1	0	...	0	0	0	1	0	0	0	0	0	0
dave	0	0	0	0	1	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
hasan	0	0	0	0	0	0	0	0	0	0	...	2	1	0	1	0	0	0	0	0	0
jim	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
joe	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
john	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	1
louis	0	0	0	0	0	3	0	0	0	0	...	0	0	0	2	0	0	0	0	0	0
mike	0	0	0	0	0	0	0	0	0	0	...	0	0	2	1	0	0	0	0	0	0
ricky	0	0	0	0	0	0	0	0	1	1	...	0	0	0	0	0	0	0	0	1	0

12 rows × 7484 columns

In [23]:
```
# Let's pickle it for later use
data_dtm.to_pickle("dtm.pkl")
```

In [24]:
```
# Let's also pickle the cleaned data (before we put it in document-term matrix format) and the CountVectorizer object
data_clean.to_pickle('data_clean.pkl')
pickle.dump(cv, open("cv.pkl", "wb"))
```

Additional Exercises

1. Can you add an additional regular expression to the clean_text_round2 function to further clean the text?
2. Play around with CountVectorizer's parameters. What is ngram_range? What is min_df and max_df?

In [ ]:
```

```