Week 1: Explore the BBC News archive

Welcome! In this assignment you will be working with a variation of the BBC News Classification Dataset, which contains 2225 examples of news articles with their respective categories.

TIPS FOR SUCCESSFUL GRADING OF YOUR ASSIGNMENT:

- All cells are frozen except for the ones where you need to submit your solutions or when explicitly mentioned you can interact with it.
- You can add new cells to experiment but these will be omitted by the grader, so don't rely on newly created cells to host your solution code, use the provided places for this.
- You can add the comment # grade-up-to-here in any graded cell to signal the grader that it must only evaluate up to that point. This is helpful if you want to check if you are on the right track even if you are not done with the whole assignment. Be sure to remember to delete the comment afterwards!
- Avoid using global variables unless you absolutely have to. The grader tests your code in an isolated environment without running all cells from the top. As a result, global variables may be unavailable when scoring your submission. Global variables that are meant to be used will be defined in UPPERCASE.
- To submit your notebook, save it and then click on the blue submit button at the beginning of the page.

Let's get started!

In [1]:

```python
import csv
import pandas as pd
import numpy as np
import tensorflow as tf
```

In [2]:

```python
import unittests
```

Begin by looking at the structure of the csv that contains the data:

In [3]:

```python
with open("./data/bbc-text.csv", 'r') as csvfile:
    print(f"First line (header) looks like this:\n\n{csvfile.readline()}")
    print(f"Each data point looks like this:\n\n{csvfile.readline()}")
```

```
First line (header) looks like this:

category,text

Each data point looks like this:

tech,tv future in the hands of viewers with home theatre systems  plasma high-definition tvs  and digital video recorders moving into th
e living room  the way people watch tv will be radically different in five years  time.  that is according to an expert panel which gath
ered at the annual consumer electronics show in las vegas to discuss how these new technologies will impact one of our favourite pastime
s. with the us leading the trend  programmes and other content will be delivered to viewers via home networks  through cable  satellite
telecoms companies  and broadband service providers to front rooms and portable devices.  one of the most talked-about technologies of c
es has been digital and personal video recorders (dvr and pvr). these set-top boxes  like the us s tivo and the uk s sky+ system  allow
people to record  store  play  pause and forward wind tv programmes when they want.  essentially  the technology allows for much more pe
rsonalised tv. they are also being built-in to high-definition tv sets  which are big business in japan and the us  but slower to take o
ff in europe because of the lack of high-definition programming. not only can people forward wind through adverts  they can also forget
about abiding by network and channel schedules  putting together their own a-la-carte entertainment. but some us networks and cable and
satellite companies are worried about what it means for them in terms of advertising revenues as well as  brand identity  and viewer loy
alty to channels. although the us leads in this technology at the moment  it is also a concern that is being raised in europe  particula
rly with the growing uptake of services like sky+.  what happens here today  we will see in nine months to a years  time in the uk   ada
m hume  the bbc broadcast s futurologist told the bbc news website. for the likes of the bbc  there are no issues of lost advertising re
venue yet. it is a more pressing issue at the moment for commercial uk broadcasters  but brand loyalty is important for everyone.  we wi
ll be talking more about content brands rather than network brands   said tim hanlon  from brand communications firm starcom mediavest.
the reality is that with broadband connections  anybody can be the producer of content.  he added:  the challenge now is that it is hard
to promote a programme with so much choice.   what this means  said stacey jolna  senior vice president of tv guide tv group  is that th
e way people find the content they want to watch has to be simplified for tv viewers. it means that networks  in us terms  or channels c
ould take a leaf out of google s book and be the search engine of the future  instead of the scheduler to help people find what they wan
t to watch. this kind of channel model might work for the younger ipod generation which is used to taking control of their gadgets and w
hat they play on them. but it might not suit everyone  the panel recognised. older generations are more comfortable with familiar schedu
les and channel brands because they know what they are getting. they perhaps do not want so much of the choice put into their hands  mr
hanlon suggested.  on the other end  you have the kids just out of diapers who are pushing buttons already - everything is possible and
available to them   said mr hanlon.  ultimately  the consumer will tell the market they want.   of the 50 000 new gadgets and technologi
es being showcased at ces  many of them are about enhancing the tv-watching experience. high-definition tv sets are everywhere and many
new models of lcd (liquid crystal display) tvs have been launched with dvr capability built into them  instead of being external boxes.
one such example launched at the show is humax s 26-inch lcd tv with an 80-hour tivo dvr and dvd recorder. one of the us s biggest satel
lite tv companies  directtv  has even launched its own branded dvr at the show with 100-hours of recording capability  instant replay  a
nd a search function. the set can pause and rewind tv for up to 90 hours. and microsoft chief bill gates announced in his pre-show keyno
te speech a partnership with tivo  called tivotogo  which means people can play recorded programmes on windows pcs and mobile devices. a
ll these reflect the increasing trend of freeing up multimedia so that people can watch what they want  when they want.
```

As you can see, each data point is composed of the category of the news article followed by a comma and then the actual text of the article.

Exercise 1: parse_data_from_file

The csv is a very common format to store data and you will probably encounter it many times so it is good to be comfortable with it. Your first exercise will be to read the data from the raw csv file so you can analyze it and built models around it. To do so, complete the `parse_data_from_file` function below.

Since this format is so common there are a lot of ways to deal with this files using python, both using the standard library or third party libraries such as pandas. Because of this the implementation details are entirely up to you, **the only requirement is that your function returns the `sentences` and `labels` as regular python lists**.

Hints:

- Remember the file contains headers so take this into consideration.
- If you are unfamiliar with libraries such as pandas or numpy and you prefer to use python's standard library, take a look at `csv.reader`, which lets you iterate over the lines of a csv file.
- You can use the `read_csv` function from the pandas library.
- You can use the `loadtxt` function from the numpy library.
- If you use any of the two latter approaches remember you still need to convert the `sentences` and `labels` to regular python lists, so take a look at the docs to see how it can be done.

In [10]:

```
# GRADED FUNCTION: parse_data_from_file

def parse_data_from_file(filename):
    """
    Extracts sentences and labels from a CSV file

    Args:
        filename (str): path to the CSV file

    Returns:
        (list[str], list[str]): tuple containing lists of sentences and labels
    """
    sentences = []
    labels = []

    ### START CODE HERE ###

    data = pd.read_csv(filename)
    sentences = data['category'].to_list()
    labels = data['text'].to_list()

    ### END CODE HERE ###

    return sentences, labels
```

In [12]:

```
# Get sentences and labels as python lists
sentences, labels = parse_data_from_file("./data/bbc-text.csv")

print(f"There are {len(sentences)} sentences in the dataset.\n")
print(f"First sentence has {len(sentences[0].split())} words.\n")
print(f"There are {len(labels)} labels in the dataset.\n")
print(f"The first 5 labels are {labels[:5]}\n\n")
```

```
There are 2225 sentences in the dataset.

First sentence has 1 words.

There are 2225 labels in the dataset.

The first 5 labels are ['tv future in the hands of viewers with home theatre systems  plasma high-definition tvs  and digital video reco
rders moving into the living room  the way people watch tv will be radically different in five years  time.  that is according to an exp
ert panel which gathered at the annual consumer electronics show in las vegas to discuss how these new technologies will impact one of o
ur favourite pastimes. with the us leading the trend  programmes and other content will be delivered to viewers via home networks  throu
gh cable  satellite  telecoms companies  and broadband service providers to front rooms and portable devices.  one of the most talked-ab
out technologies of ces has been digital and personal video recorders (dvr and pvr). these set-top boxes  like the us s tivo and the uk
s sky+ system  allow people to record  store  play  pause and forward wind tv programmes when they want.  essentially  the technology al
lows for much more personalised tv. they are also being built-in to high-definition tv sets  which are big business in japan and the us
but slower to take off in europe because of the lack of high-definition programming. not only can people forward wind through adverts  t
hey can also forget about abiding by network and channel schedules  putting together their own a-la-carte entertainment. but some us net
works and cable and satellite companies are worried about what it means for them in terms of advertising revenues as well as  brand iden
tity  and viewer loyalty to channels. although the us leads in this technology at the moment  it is also a concern that is being raised
in europe  particularly with the growing uptake of services like sky+.  what happens here today  we will see in nine months to a years
time in the uk   adam hume  the bbc broadcast s futurologist told the bbc news website. for the likes of the bbc  there are no issues of
lost advertising revenue yet. it is a more pressing issue at the moment for commercial uk broadcasters  but brand loyalty is important f
or everyone.  we will be talking more about content brands rather than network brands   said tim hanlon  from brand communications firm
starcom mediavest.  the reality is that with broadband connections  anybody can be the producer of content.  he added:  the challenge no
w is that it is hard to promote a programme with so much choice.   what this means  said stacey jolna  senior vice president of tv guide
tv group  is that the way people find the content they want to watch has to be simplified for tv viewers. it means that networks  in us
terms  or channels could take a leaf out of google s book and be the search engine of the future  instead of the scheduler to help peopl
e find what they want to watch. this kind of channel model might work for the younger ipod generation which is used to taking control of
their gadgets and what they play on them. but it might not suit everyone  the panel recognised. older generations are more comfortable w
ith familiar schedules and channel brands because they know what they are getting. they perhaps do not want so much of the choice put in
to their hands  mr hanlon suggested.  on the other end  you have the kids just out of diapers who are pushing buttons already - everythi
ng is possible and available to them   said mr hanlon.  ultimately  the consumer will tell the market they want.   of the 50 000 new gad
gets and technologies being showcased at ces  many of them are about enhancing the tv-watching experience. high-definition tv sets are e
verywhere and many new models of lcd (liquid crystal display) tvs have been launched with dvr capability built into them  instead of bei
ng external boxes. one such example launched at the show is humax s 26-inch lcd tv with an 80-hour tivo dvr and dvd recorder. one of the
us s biggest satellite tv companies  directtv  has even launched its own branded dvr at the show with 100-hours of recording capability
instant replay  and a search function. the set can pause and rewind tv for up to 90 hours. and microsoft chief bill gates announced in h
is pre-show keynote speech a partnership with tivo  called tivotogo  which means people can play recorded programmes on windows pcs and
mobile devices. all these reflect the increasing trend of freeing up multimedia so that people can watch what they want  when they want.
', 'worldcom boss  left books alone  former worldcom boss bernie ebbers  who is accused of overseeing an $11bn (£5.8bn) fraud  never mad
e accounting decisions  a witness has told jurors.  david myers made the comments under questioning by defence lawyers who have been arg
uing that mr ebbers was not responsible for worldcom s problems. the phone company collapsed in 2002 and prosecutors claim that losses w
ere hidden to protect the firm s shares. mr myers has already pleaded guilty to fraud and is assisting prosecutors.  on monday  defence
lawyer reid weingarten tried to distance his client from the allegations. during cross examination  he asked mr myers if he ever knew mr
ebbers  make an accounting decision  .  not that i am aware of   mr myers replied.  did you ever know mr ebbers to make an accounting en
try into worldcom books   mr weingarten pressed.  no   replied the witness. mr myers has admitted that he ordered false accounting entri
es at the request of former worldcom chief financial officer scott sullivan. defence lawyers have been trying to paint mr sullivan  who
has admitted fraud and will testify later in the trial  as the mastermind behind worldcom s accounting house of cards.  mr ebbers  team
meanwhile  are looking to portray him as an affable boss  who by his own admission is more pe graduate than economist. whatever his abil
ities  mr ebbers transformed worldcom from a relative unknown into a $160bn telecoms giant and investor darling of the late 1990s. world
com s problems mounted  however  as competition increased and the telecoms boom petered out. when the firm finally collapsed  shareholde
rs lost about $180bn and 20 000 workers lost their jobs. mr ebbers  trial is expected to last two months and if found guilty the former
ceo faces a substantial jail sentence. he has firmly declared his innocence.', 'tigers wary of farrell  gamble  leicester say they will
not be rushed into making a bid for andy farrell should the great britain rugby league captain decide to switch codes.   we and anybody
else involved in the process are still some way away from going to the next stage   tigers boss john wells told bbc radio leicester.  at
the moment  there are still a lot of unknowns about andy farrell  not least his medical situation.  whoever does take him on is going to
take a big  big gamble.  farrell  who has had persistent knee problems  had an operation on his knee five weeks ago and is expected to b
e out for another three months. leicester and saracens are believed to head the list of rugby union clubs interested in signing farrell
if he decides to move to the 15-man game.  if he does move across to union  wells believes he would better off playing in the backs  at
least initially.  i m sure he could make the step between league and union by being involved in the centre   said wells.  i think englan
d would prefer him to progress to a position in the back row where they can make use of some of his rugby league skills within the forwa
rds.  the jury is out on whether he can cross that divide.  at this club  the balance will have to be struck between the cost of that ga
mble and the option of bringing in a ready-made replacement.', 'yeading face newcastle in fa cup premiership side newcastle united face
a trip to ryman premier league leaders yeading in the fa cup third round.  the game - arguably the highlight of the draw - is a potentia
l money-spinner for non-league yeading  who beat slough in the second round. conference side exeter city  who knocked out doncaster on s
aturday  will travel to old trafford to meet holders manchester united in january. arsenal were drawn at home to stoke and chelsea will
play host to scunthorpe. the only other non-league side in the draw are hinckley united  who held brentford to a goalless draw on sunday
. they will meet league one leaders luton if they win their replay against martin allen s team at griffin park.  a number of premiership
teams face difficult away games against championship sides on the weekend of 8/9 january. third-placed everton visit plymouth  liverpool
travel to burnley  crystal palace go to sunderland  fulham face carling cup semi-finalists watford  bolton meet ipswich  while aston vil
la were drawn against sheffield united. premiership strugglers norwich  blackburn  west brom are away at west ham  cardiff and preston n
orth end respectively. southampton visit northampton  having already beaten the league two side in the carling cup earlier this season.
middlesbrough were drawn away against either swindon or notts county  while spurs entertain brighton at white hart lane.  arsenal v stok
e  swindon/notts co v middlesbrough  man utd v exeter  plymouth v everton  leicester v blackpool  derby v wigan  sunderland v crystal pa
lace  wolves v millwall  yeading v newcastle  hull v colchester  tottenham v brighton  reading v stockport/swansea  birmingham v leeds
hartlepool v boston  milton keynes dons v peterborough  oldham v man city  chelsea v scunthorpe  cardiff v blackburn  charlton v rochdal
e  west ham v norwich  sheff utd v aston villa  preston v west brom  rotherham v yeovil  burnley v liverpool  bournemouth v chester  cov
entry v crewe  watford v fulham  ipswich v bolton  portsmouth v gillingham  northampton v southampton  qpr v nottm forest  luton v hinck
ley/brentford  matches to be played on weekend of 8/9 january.', 'ocean s twelve raids box office ocean s twelve  the crime caper sequel
starring george clooney  brad pitt and julia roberts  has gone straight to number one in the us box office chart.  it took $40.8m (£21m)
in weekend ticket sales  according to studio estimates. the sequel follows the master criminals as they try to pull off three major heis
ts across europe. it knocked last week s number one  national treasure  into third place. wesley snipes  blade: trinity was in second  t
aking $16.1m (£8.4m). rounding out the top five was animated fable the polar express  starring tom hanks  and festive comedy christmas w
ith the kranks.  ocean s twelve box office triumph marks the fourth-biggest opening for a december release in the us  after the three fi
lms in the lord of the rings trilogy. the sequel narrowly beat its 2001 predecessor  ocean s eleven which took $38.1m (£19.8m) on its op
ening weekend and $184m (£95.8m) in total. a remake of the 1960s film  starring frank sinatra and the rat pack  ocean s eleven was direc
ted by oscar-winning director steven soderbergh. soderbergh returns to direct the hit sequel which reunites clooney  pitt and roberts wi
th matt damon  andy garcia and elliott gould. catherine zeta-jones joins the all-star cast.  it s just a fun  good holiday movie   said
dan fellman  president of distribution at warner bros. however  us critics were less complimentary about the $110m (£57.2m) project  wit
h the los angeles times labelling it a  dispiriting vanity project . a milder review in the new york times dubbed the sequel  unabashedl
y trivial .']
```

Expected Output:

```
There are 2225 sentences in the dataset.

First sentence has 737 words.

There are 2225 labels in the dataset.

The first 5 labels are ['tech', 'business', 'sport', 'sport', 'entertainment']
```

In [13]:

```python
# Test your code!
unittests.test_parse_data_from_file(parse_data_from_file)
```

```
All tests passed!
```

An important note:

At this point you typically would convert your data into a `tf.data.Dataset` (alternatively you could have used tf.data.experimental.CsvDataset to do this directly but since this is an experimental feature it is better to avoid when possible) but for this assignment you will keep working with the data as regular python lists.

The reason behind this is that by using a `tf.data.Dataset` some parts of this assignment will be much more difficult (in particular the next exercise), because when dealing with tensors you need to take additional considerations that you don't need to when dealing with lists and since this is the first assignment of the course, it is best to keep things simple. During next week's assignment you will get to see how this process looks like but for now carry on with the data in this format and worry not since TensorFlow is still compatible with these data formats!

Exercise 2: standardize_func

One important step when working with text data is to standardize it so it is easier to extract information out of it. For instance, you probably want to convert it all to lower-case (so the same word doesn't have different representations such as "hello" and "Hello") and to remove the stopwords from it. These are the most common words in the language and they rarely provide useful information for the classification process. The next cell provides a list of common stopwords which you can use in the exercise:

In [14]:

```python
# List of stopwords
STOPWORDS = ["a", "about", "above", "after", "again", "against", "all", "am", "an", "and", "any", "are", "as", "at", "be", "because", "
```

To achieve this, complete the `standardize_func` below. This function should receive a string and return another string that excludes all of the stopwords provided from it, as well as converting it to lower-case.

Hints:

- You only need to account for whitespace as the separation mechanism between words in the sentence.
- The list of stopwords is already provided for you as a global variable you can safely use.
- Check out the lower method for python strings.
- The returned sentence should not include extra whitespace so the string "hello   again   FRIENDS" should be standardized to "hello friends".

In [38]:

```python
# GRADED FUNCTION: standardize_func

def standardize_func(sentence):
    """Standardizes sentences by converting to lower-case and removing stopwords.

    Args:
        sentence (str): Original sentence.

    Returns:
        str: Standardized sentence in lower-case and without stopwords.
    """

    ### START CODE HERE ###
    words = sentence.lower().split()
    filtered_words = []
    for word in words:
        if word not in STOPWORDS:
            filtered_words.append(word)

    ### END CODE HERE ###
    return ' '.join(filtered_words)
```

In [39]:
```
test_sentence = "Hello! We're just about to see this function in action =)"
standardized_sentence = standardize_func(test_sentence)
print(f"Original sentence is:\n{test_sentence}\n\nAfter standardizing:\n{standardized_sentence}")

standard_sentences = [standardize_func(sentence) for sentence in sentences]

print("\n\n--- Apply the standardization to the dataset ---\n")
print(f"There are {len(standard_sentences)} sentences in the dataset.\n")
print(f"First sentence has {len(sentences[0].split())} words originally.\n")
print(f"First sentence has {len(standard_sentences[0].split())} words (after removing stopwords).\n")
```

```
Original sentence is:
Hello! We're just about to see this function in action =)

After standardizing:
hello! just see function action =)


--- Apply the standardization to the dataset ---

There are 2225 sentences in the dataset.

First sentence has 1 words originally.

First sentence has 1 words (after removing stopwords).
```

Expected Output:

```
Original sentence is:
Hello! We're just about to see this function in action =)

After standardizing:
hello! just see function action =)


--- Apply the standardization to the dataset ---

There are 2225 sentences in the dataset.

First sentence has 737 words originally.

First sentence has 436 words (after removing stopwords).
```

In [40]:
```
# Test your code!
unittests.test_standardize_func(standardize_func)
```

```
All tests passed!
```

With the dataset standardized you could go ahead and convert it to a `tf.data.Dataset` , which you will NOT be doing for this assignment. However if you are curious, this can be achieved like this:

```
dataset = tf.data.Dataset.from_tensor_slices((standard_sentences, labels))
```

Exercise 3: fit_vectorizer

Now that your data is standardized, it is time to vectorize the sentences of the dataset. For this complete the `fit_vectorizer` below.

This function should receive the list of sentences as input and return a `tf.keras.layers.TextVectorization` that has been adapted to those sentences.

In [43]:
```
# GRADED FUNCTION: fit_vectorizer

def fit_vectorizer(sentences):
    """
    Instantiates the TextVectorization layer and adapts it to the sentences.

    Args:
        sentences (list[str]): lower-cased sentences without stopwords

    Returns:
        tf.keras.layers.TextVectorization: an instance of the TextVectorization layer adapted to the texts.
    """

    ### START CODE HERE ###

    # Instantiate the TextVectorization class
    vectorizer = tf.keras.layers.TextVectorization()

    vectorizer.adapt(sentences)

    # Adapt to the sentences


    ### END CODE HERE ###

    return vectorizer
```

In [44]:
```
# Create the vectorizer adapted to the standardized sentences
vectorizer = fit_vectorizer(standard_sentences)

# Get the vocabulary
vocabulary = vectorizer.get_vocabulary()

print(f"Vocabulary contains {len(vocabulary)} words\n")
print("[UNK] token included in vocabulary" if "[UNK]" in vocabulary else "[UNK] token NOT included in vocabulary")
```

```
Vocabulary contains 7 words

[UNK] token included in vocabulary
```

Expected Output:

```
Vocabulary contains 33088 words

[UNK] token included in vocabulary
```

In [45]:
```
# Test your code!
unittests.test_fit_vectorizer(fit_vectorizer)
```

```
All tests passed!
```

Next, you can use the adapted vectorizer to vectorize the sentences in your dataset. Notice that by default `tf.keras.layers.TextVectorization` pads the sequences so all of them have the same length (typically the length of the longest sentence will be used if no truncation is defined), this is important because neural networks expect the inputs to have the same size.

In [52]:
```
# Vectorize and pad sentences
padded_sequences = vectorizer(standard_sentences)

# Show the output
print(f"First padded sequence looks like this: \n\n{padded_sequences[0]}\n")
print(f"Tensor of all sequences has shape: {padded_sequences.shape}\n")
print(f"This means there are {padded_sequences.shape[0]} sequences in total and each one has a size of {padded_sequences.shape[1]}")
```

```
First padded sequence looks like this:

[5]

Tensor of all sequences has shape: (2225, 1)

This means there are 2225 sequences in total and each one has a size of 1
```

Notice that now the variable refers to `sequences` rather than `sentences`. This is because all your text data is now encoded as a sequence of integers.

Exercise 4: fit_label_encoder

With the sentences already vectorized it is time to encode the labels so they can also be fed into a neural network. For this complete the `fit_label_encoder` below.

This function should receive the list of labels as input and return a `tf.keras.layers.StringLookup` that has been adapted to those sentences. In theory you could also use `tf.keras.layers.TextVectorization` layer here but it provides a lot of extra functionality that is not required so it ends up being overkill.

`tf.keras.layers.StringLookup` is able to perform the job just fine and it is much simpler.

Hints:

- Since all of the texts have their corresponding labels you need to ensure that the vocabulary does not include the out-of-vocabulary (OOV) token since that is not a valid label.

In [65]:
```
# GRADED FUNCTION: fit_label_encoder

def fit_label_encoder(labels):
    """
    Tokenizes the labels
    Args:
        labels (list[str]): labels to tokenize

    Returns:
        tf.keras.layers.StringLookup: adapted encoder for labels
    """
    ### START CODE HERE ###

    # Instantiate the StringLookup layer. Remember that you don't want any OOV tokens!
    label_encoder = tf.keras.layers.StringLookup(num_oov_indices=0)

    # Adapt the StringLookup layer to the labels
    label_encoder.adapt(labels)

    ### END CODE HERE ###

    return label_encoder
```

In [66]:

```
# Create the encoder adapted to the labels
label_encoder = fit_label_encoder(labels)

# Get the vocabulary
vocabulary = label_encoder.get_vocabulary()

# Encode labels
label_sequences = label_encoder(labels)

print(f"Vocabulary of labels looks like this: {vocabulary}\n")
print(f"First ten labels: {labels[:10]}\n")
print(f"First ten label sequences: {label_sequences[:10]}\n")
```

```
IOPub data rate exceeded.
The Jupyter server will temporarily stop sending output
to the client in order to avoid crashing it.
To change this limit, set the config variable
`--ServerApp.iopub_data_rate_limit`.

Current values:
ServerApp.iopub_data_rate_limit=1000000.0 (bytes/sec)
ServerApp.rate_limit_window=3.0 (secs)
```

Expected Output:

```
Vocabulary of labels looks like this: ['sport', 'business', 'politics', 'tech', 'entertainment']

First ten labels: ['tech', 'business', 'sport', 'sport', 'entertainment', 'politics', 'politics', 'sport', 'sport',
'entertainment']

First ten label sequences: [3 1 0 0 4 2 2 0 0 4]
```

You should see that each encoded label corresponds to the index of its corresponding label in the vocabulary!

In [67]:

```
# Test your code!
unittests.test_fit_label_encoder(fit_label_encoder)
```

```
All tests passed!
```

Great job! Now you have successfully performed all the necessary steps to train a neural network capable of processing text. This is all for now but in next week's assignment you will train a model capable of classifying the texts in this same dataset!

Congratulations on finishing this week's assignment!

You have successfully implemented functions to process various text data processing ranging from pre-processing, reading from raw files and tokenizing text.

Keep it up!