Strategy for ML Project

1) **Preprocessing of the text**

First we get rid of numbers, punctuations and extra spaces in the text string. ---- DONE

Then we need to remove the stop words such as “the, an, and etc.” so that we get a dictionary of words. I need to get a whole list of all such words online and remove from my text.

**2) Tokenization**

We create a bag of words which is just a dictionary of the words and their frequencies with which they appear in a text string line. --- DONE

For example “How are you my friend?” when tokenized becomes a list of words as:

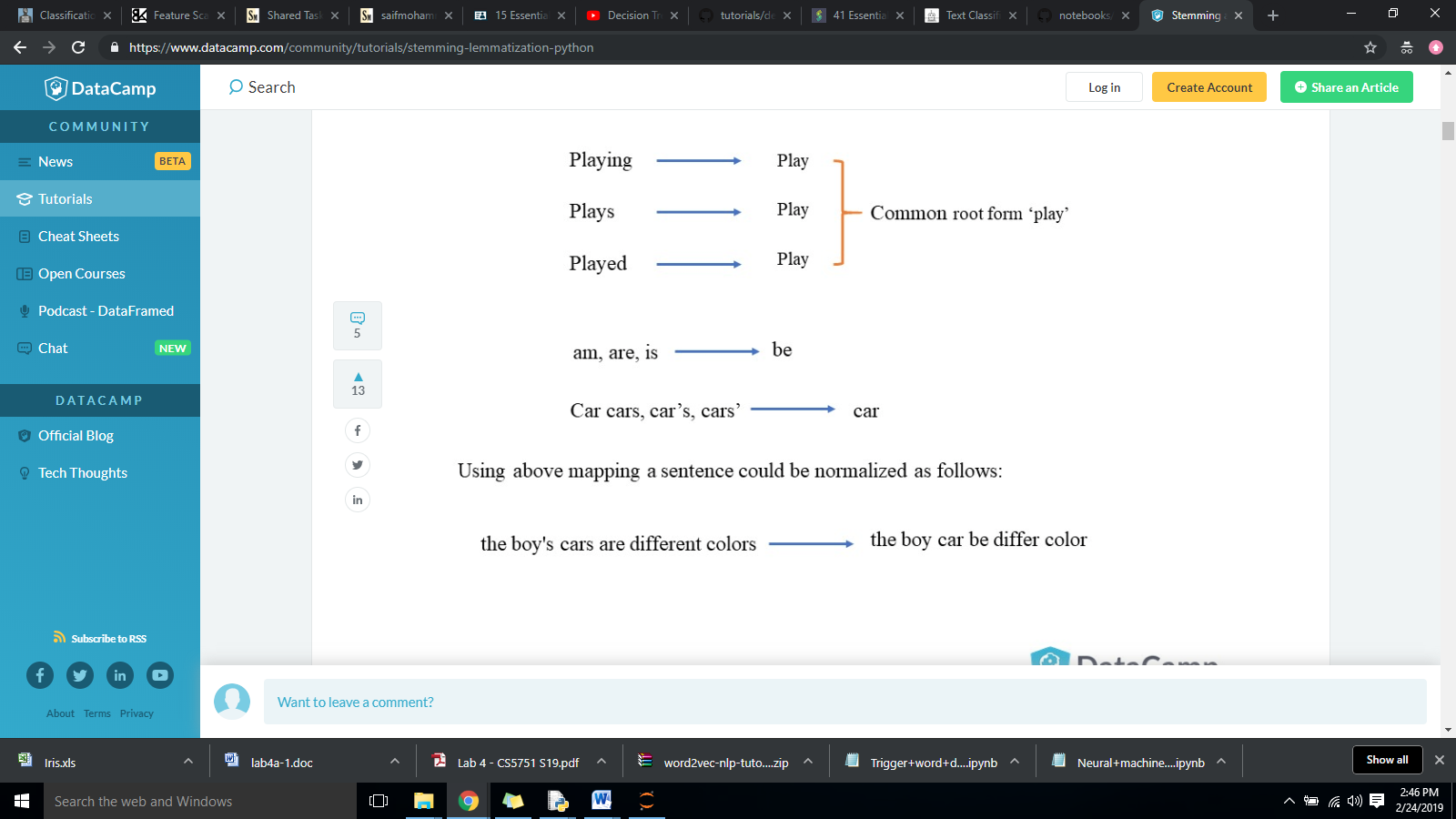
[‘How’,’are’,’you’,’my’,’friend’,’?’]

**3) Classifier**

Now we create a classifier using Neural networks for which we will start with a single hidden layer RELU activation function neural network with dimensions equal to (max length of tweet) x (training samples). And see what results we get. The last layer will be of softmax activation function with 4 units as we have 4 classes.

### Background

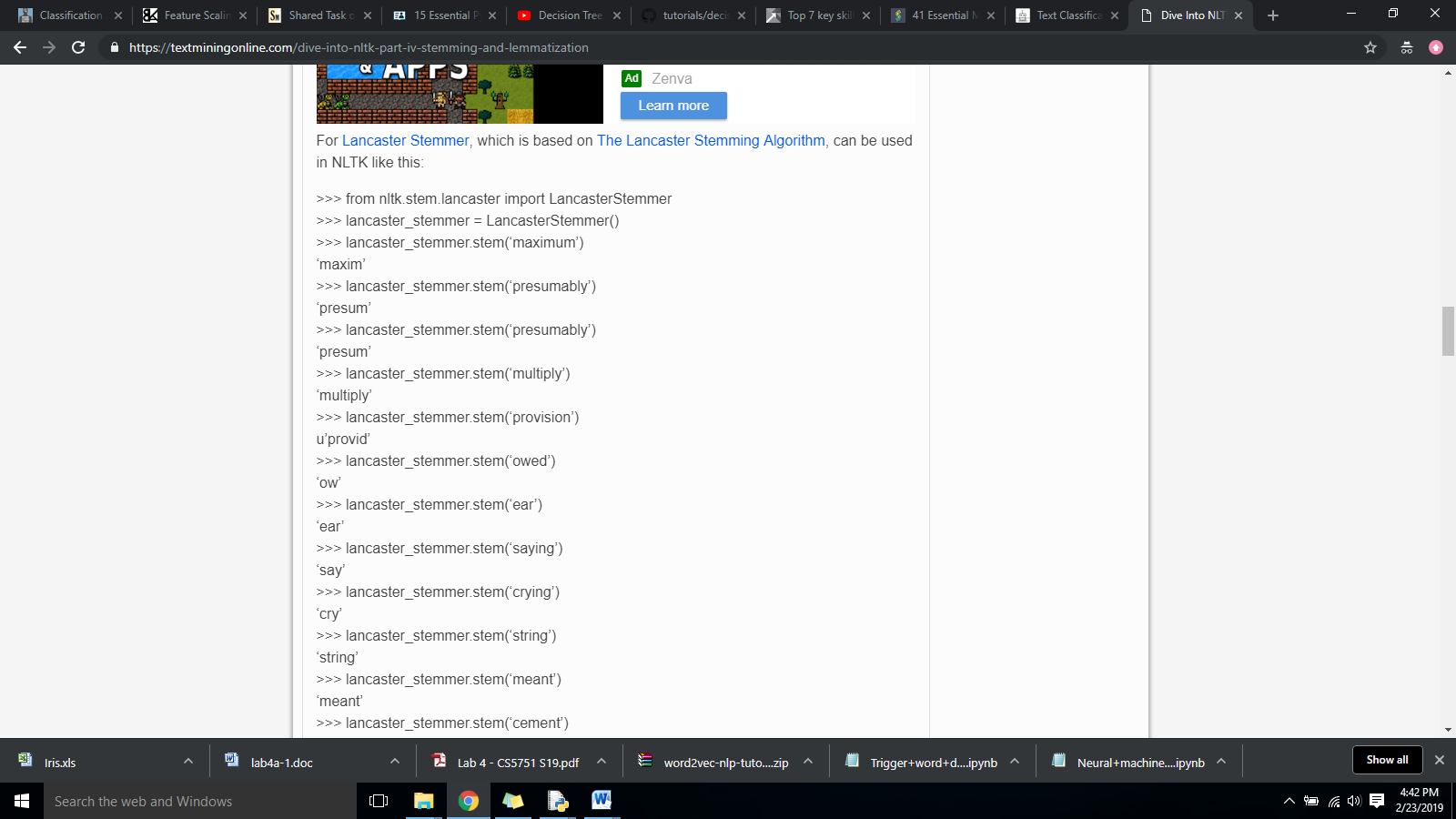
Languages we speak and write are made up of several words often derived from one another. When a language contains words that are derived from another word as their use in the speech changes is called Inflected Language.



Stemming and Lemmatization helps us to achieve the root forms (sometimes called synonyms in search context) of inflected (derived) words. We use Natural Language Toolkit NLTK which has Porter and Lancaster Stemmer.

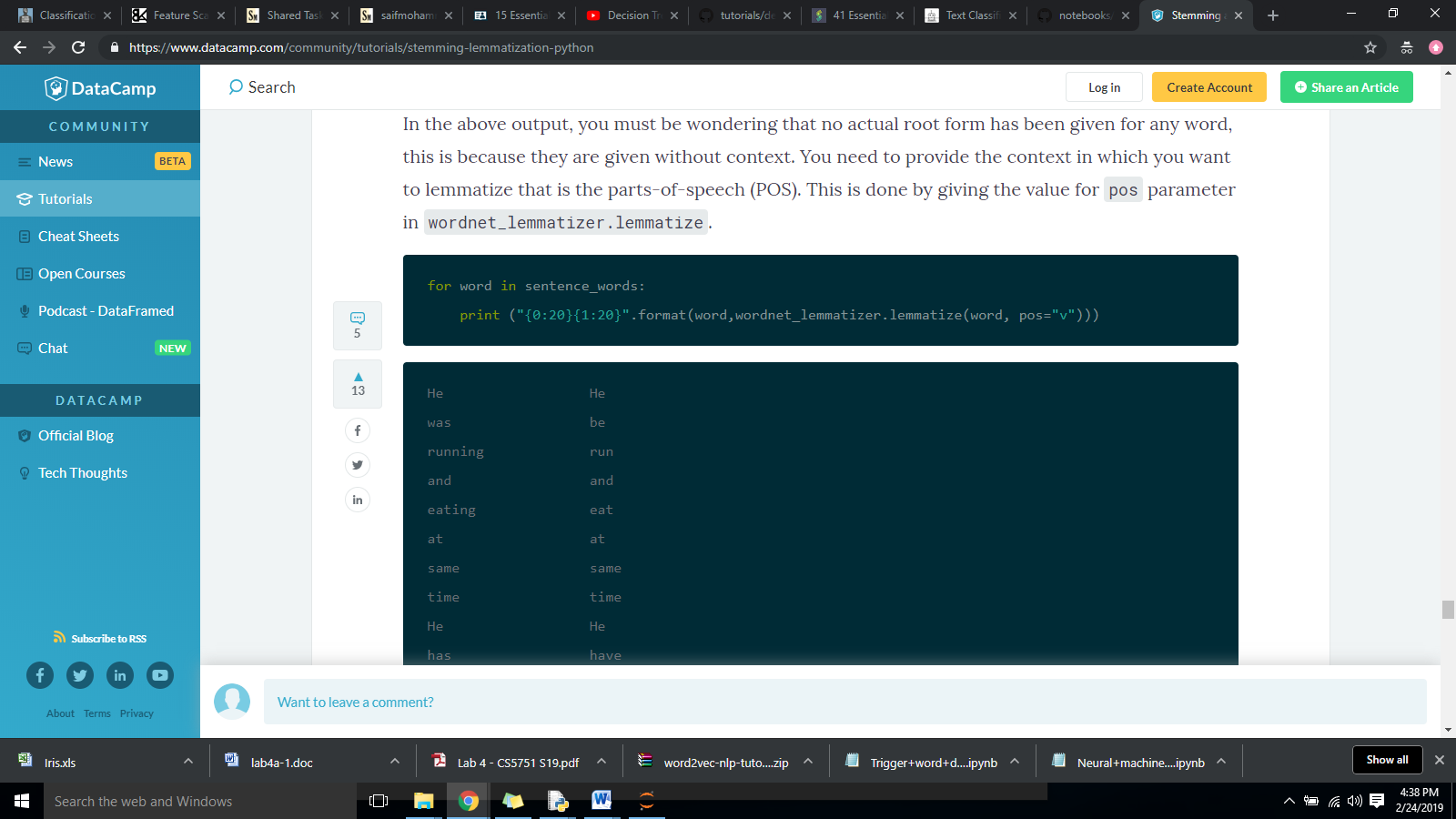
### Stemming

Stemming in text processing means cutting the word to its origin for example, walking walked, walker have stem “**walk**”. NLTK toolkit in Python has different kinds of stemmers such as Porter, Lancaster stemmer etc,



### Lemmatization

Lemmatization, unlike Stemming, reduces the inflected words properly ensuring that the root word belongs to the language. In Lemmatization root word is called Lemma. A lemma (plural lemmas or lemmata) is the canonical form, dictionary form, or citation form of a set of words.



### Stemming or Lemmatization

* Stemming and Lemmatization both generate the root form of the inflected words. The difference is that stem might not be an actual word whereas, lemma is an actual language word.
* Stemming follows an algorithm with steps to perform on the words which makes it faster. Whereas, in lemmatization, you used WordNet corpus and a corpus for stop words as well to produce lemma which makes it slower than stemming. You also had to define a parts-of-speech to obtain the correct lemma.

### Information about dataset

<http://saifmohammad.com/WebPages/EmotionIntensity-SharedTask.html>

This link has 4 kinds of dataset: one for anger, one for fear, one for joy and one for sadness for training set and 4 files for test set. Each of these datasets have the following format:

All sentences start with a

[number, tweet\_sentence, label [fear,anger,joy,sadness], intensity\_of\_emotion]

The intensity of emotion varies from 0 to 1 where 1 expresses high intensity of anger, fear, joy or sadness. It is seen that only tweets with threshold higher than 0.6 “ACTUALLY” express the sentiment.

We are keeping this threshold of 0.6 which leaves us with the following:

**Anger\_training\_set: emotion intensity threshold 0.5 411 lines**

**Fear\_training\_set: emotion intensity threshold 0.6 347 lines**

**Joy\_training\_set: emotion intensity threshold 0.5 405 lines**

**Sadness\_training\_set: emotion intensity threshold 0.5 384 lines**

**Total we have 1547 lines in training data**

**And total 4895 unique words that include emoticons but exclude**