5th Assignment

NLTK Library – Sentence Generator & Classify Reviews

Machine Learning and Natural Language Processing

Christakakis Panagiotis

ID: aid23004

Master of Science in
Artificial Intelligence
and Data Analytics

University of Macedonia
School of Information Sciences
Department of Applied Informatics

Postgraduate Studies Program in "Artificial Intelligence and Data Analytics", University Of Macedonia

Date: 09/01/2023

Unsupervised Learning - Clustering

Contents

1 Introduction

In this assignment we worked with NLTK library and the two parts were:

1) Sentence creation from bigrams and trigrams generated by Project Gutenberg books.
2) Train a classifier to find positive and negative movie reviews

For implementing our methods we’ll use python and appropriate libraries. The experiments run on CPU Intel Core i9-10900K, RAM 128GB DDR4, GPU RTX3090.

2 Methods

(A)

Firstly, 10 books were chosen and combined into a single large text file. Each sentence of this file was tokenized and afterwards each one of them was split into single word tokens. A list of unigrams, bigrams, trigrams were created as well as their frequencies. Each frequency list was split into tuples in order to have the information-words we needed in key – value like format.

Two separate functions were created, in order to generate the final sentences. For bigram sentences the first token was randomly chosen, only from tokens that were real words, i.e. a-z and A-Z. The full stop was picked up as an end token. The function for trigram sentences contained an extra for loop that was necessary to find all the third words that were already chosen. As starting tokens, the words ‘It is’ were chosen and again full stop was the end token. Afterwards these functions were called to create our 20 final sentences.

Sentences created with bigrams:

1) brick house while now offering -- If only intent In Tennessee and inscriptions stamped up half falling head suspends The loaded barges , invested form to water hailed a sweetener
2) the reeling scene that spoke -- sang out : 'Tis now beside a clearness , ride on whale started the Spotted Dog , LEATHER STRAPS , latent , courses over
3) where the Squid .
4) happily foreknowing may escape was fastened at innocent or herd the uncanonical Rabbins , Tempt him live would yesterday I received very distance the Milky Way .
5) had selected the token of unimaginable casualty , belou 'd hurricanes , withhold the intercourse .
6) fifth act is Satan , quietly digesting and indiscriminately befallen any allusion to show of address ! overboard ! keep of wildness as they who gave the outblown rumors and
7) the slim fellow no Jackal kindly feeling heart gave glimpses here patiently assured , get better home too unwell .
8) his boys usually turned crimson with spices , shelter her manner established in pulses American actor with indulgence ! Yet must save thee home : good-bye and told me faint
9) every sort in acknowledgement of noiseless around of inconsistency in YOUR sake if lifted ; suspended in disgust at what nameless atrocity .
10) Thrasher than attraction .

Sentences created with trigrams:

1) It is that good preposterous one ever the over sitting sign generally talking closest put to an bald just also you right rude fastidious all full unpleasant there time so.
2) It is n't really really readily so the not sufficient but Goliath common little not the that provided a to wild gone without the feeding for called not n't heard
3) It is truly at Louisa on it only it dust poor bequeathing deep a partially a one college Fanny in this in , after this large first all the coming
4) It is to used dead jocund your no Colonel erect in now right the nothing the perpendicularly all this heightened young to but never just I not all , Moby
5) It is dismissed clearer locked very drawing a mostly no quite your autumn certain anchor great equally , -- just on sacred for smaller free but rather , in this
6) It is like strong an actually dropt mine in become good generally to nearly as the , caused) a the a n't a in the , what .
7) It is fit made truth no heard invested always Mr. short much a proofe different learning blown true shot terrible. somewhat hotter good tough privileged one very perpetual without to
8) It is a to determined she almost another Mr begun this useless not tough Seal its proved ? an worship he to harpooned right always on the your I very
9) It is intolerable advertised safe not always every a no out of known committed still rotten your better two thine that spring cousin most not she not VERY (like
10) It is taken their worse Race rude circled necessary called my a of merely a formal , this my about dicotyledonous stove man very in possess most real sometimes spread

(B)

After downloading the package that contained the dataset with the movie reviews, a bag of words dictionary was created that was free of punctuation and words that wouldn't help in the training of the classifier. Two separate lists are created and each review gets either a positive or a negative category. Shuffling and especially stratifying is needed in our train and test data in order to maintain the same analogy of positive and negative reviews in the training and test set. Otherwise our classifier might fail to classify a category correctly, because of the proportion of the two categories.

Finally, our classifier finds correctly almost 70% of the movie reviews and also finds correctly custom-made comments like the following:

1) "This film was awful. Acting was poor and the general picture was a disaster."
2) "Loved the actors. The story was amazing and direction at its best. Wonderful movie."

For the first comment Naïve Bayes Classifier guessed that it was a negative one with 89%, while for the second one a surprisingly 94.4% predicted that it was a positive review.

3 References

1) Munck, L. D. (2021, March 15). *Mandatory assignment 1*. Deepnote. Retrieved January 8, 2023, from https://deepnote.com/@laust-dixen-munck/Mandatory-Assignment-1-61876a0e-6300-4a8b-b5a3-ab5d647e15f2

2) HaelC. (n.d.). *NLP-with-python-and-NLTK-solutions/Chapter 2.ipynb at master · HaelC/NLP-with-python-and-NLTK-solutions*. GitHub. Retrieved January 7, 2023, from https://github.com/HaelC/NLP-with-Python-and-NLTK-Solutions/blob/master/Chapter%202.ipynb

3) ODonnell, M. (2022, October 12). *NLP in python: A Primer on NLTK (with Project Gutenberg)*. Medium. Retrieved January 8, 2023, from https://odonnell31.medium.com/nlp-in-python-a-primer-on-nltk-with-project-gutenberg-fcc02be63d9a

4) Chapagain, M. (2022, October 6). *Python NLTK: Sentiment analysis on movie reviews [natural language processing (NLP)]*. Mukesh Chapagain Blog. Retrieved January 9, 2023, from https://blog.chapagain.com.np/python-nltk-sentiment-analysis-on-movie-reviews-natural-language-processing-nlp/

5) Starnes, D. (2022, May 15). *Generating sentences with n-grams using python*. YouTube. Retrieved January 7, 2023, from https://www.youtube.com/watch?v=pEYfD5aVrRI