İstanbul Bilgi University

Department of Computer Engineering

Spring 2018-2019

CMPE 346 : Natural Language Processing

Term Project - Phase II

29th March, 2019 23:59

1. The term project is composed of several phases.
2. Second phase is to investigate the language model of the collected data.
3. For each phase of the term project, you need to make a submission before the deadline.
4. After the submission of the regarding step, you need to be present during the determined lab section and make a demonstration to explain your work and to show your progress.
5. The choices of the programming language and the development environment are up to you. However, it is highly encouraged to use Pyhton and NLTK package.
6. NLTK is quite popular and widely used open source package for text analysis and natural language processing purposes.
7. You can refer to the sample programs that we have done during the lab sessions.
8. The official website of the NLTK[1] contains practical tutorials, samples and documentation which might be really beneficial for the project.
9. Please check announcements and file uploads regularly for the upcoming steps of the project.

[1] http://www.nltk.org/

CONTINUED

Steps of Phase II

1. You are expected to create a function named as **preprocess**, which takes the text as parameter, and returns the tokenized version of the text that does not contain neither any stop words nor any punctuations.

   Example:

   Text:

   Pamuk created an actual Museum of Innocence, consisting of everyday objects tied to the narrative, and housed them at an Istanbul house he purchased. Pamuk collaborated on a documentary "The Innocence Of Memories" that expanded on his Museum of Innocence. Pamuk stated that "(Museum of Dreams will) tell a different version of the love story set in Istanbul through objects and Grant Gees wonderful new film". In both Snow and the Museum of Innocence Pamuk describes tragic love-stories, where men fall in love with beautiful women at first sight. It has been noted[by whom?] that Pamuk's portrayals of women and the reasons men fall in love with them are powerful in their intensity, yet superficial in the way these love stories originate. Pamuk's heroes tend to be educated men who fall tragically in love with beauties, but who seem doomed to a decrepit loneliness.

   In 2013, Pamuk invited Grazia Toderi, whose work he admired, to design a work for the Museum of Innocence in Istanbul. Their collaboration culminated in the exhibition Words and Stars. Words and Stars opened on April 2, 2017, at the MART (Museo di Arte Moderna e Contemporanea di Trento e Rovereto), and which explores "the inclination of man to explore space and innate vocation to question the stars." The show was curated by Gianfranco Maraniello.[20] It also showed from November 4, 2016 to March 29, 2017 from 56 November 2016 at the Palazzo Madama, Piazza Castello, Turin, and at Infini-to, the Planetarium of Turin (Infini.to - Planetario di Torino, Museo dell'Astronomia e dello Spazio) by invitation.

   Resulted Text:

   ['pamuk', 'created', 'actual', 'museum', 'innocence', 'consisting', 'everyday', 'objects', 'tied', 'narrative', 'housed', 'istanb ul', 'house', 'purchased', 'pamuk', 'collaborated', 'documentary', 'innocence', 'memories', 'expanded', 'museum', 'innocence', 'p amuk', 'stated', 'museum', 'dreams', 'tell', 'different', 'version', 'love', 'story', 'set', 'istanbul', 'objects', 'grant', 'gee ', 'wonderful', 'new', 'film', 'snow', 'museum', 'innocence', 'pamuk', 'describes', 'tragic', 'men', 'fall', 'love', 'beautiful', 'women', 'first', 'sight', 'noted', 'pamuk', 'portrayals', 'women', 'reasons', 'men', 'fall', 'love', 'powerful', 'intensity', ' yet', 'superficial', 'way', 'love', 'stories', 'originate', 'pamuk', 'heroes', 'tend', 'educated', 'men', 'fall', 'tragically', ' love', 'beauties', 'seem', 'doomed', 'decrepit', 'loneliness', 'pamuk', 'invited', 'grazia', 'toderi', 'whose', 'work', 'admired' , 'design', 'work', 'museum', 'innocence', 'istanbul', 'collaboration', 'culminated', 'exhibition', 'words', 'stars', 'words', 's tars', 'opened', 'april', 'mart', 'museo', 'di', 'arte', 'moderna', 'e', 'contemporanea', 'di', 'trento', 'e', 'rovereto', 'explo res', 'inclination', 'man', 'explore', 'space', 'innate', 'vocation', 'question', 'stars', 'show', 'curated', 'gianfranco', 'maraniello', 'also', 'showed', 'november', 'march', 'november', 'palazzo', 'madama', 'piazza', 'castello', 'turin', 'planetarium', 'turin', 'planetario', 'di', 'torino', 'museo', 'e', 'dello', 'spazio', 'invitation']

2. You are expected to create a function named as **mostFrequent**, which takes tokenized version of text and a number n as parameters, and returns the number of the occurrences of the frequent words.

   Example: n = 5

   [('pamuk', 7), ('museum', 5), ('innocence', 5), ('love', 5), ('istanbul', 3)]

3. In order to have insight about the language model of the collected textual data of yours, it is aimed to make use of n-gram language models. You are required to create a function named as displayNgrams, which takes tokenized text and a number n as parameters, and displays n grams only as many as the desired n.

   Example: n = 2

   [('pamuk', 'created'),
   ('created', 'actual'),
   ('actual', 'museum'),
   ('museum', 'innocence'),
   ('innocence', 'consisting'),
   ('consisting', 'everyday'),
   ('everyday', 'objects'),
   ('objects', 'tied'), ...]

   Example: n = 3

   [('pamuk', 'created', 'actual'),
   ('created', 'actual', 'museum'),
   ('actual', 'museum', 'innocence'),
   ('museum', 'innocence', 'consisting'),
   ('innocence', 'consisting', 'everyday'),
   ('consisting', 'everyday', 'objects'), ...]

4. You are required to write a function named as **mostFreqBigram**, which takes frequency of the bigram, number of the bigrams that are going to be listed and a list of bigrams, and returns only the bigrams with the given frequency rate.In other words, you need to apply some sort of filter to ignore other frequency rate than the given frequency rate.

   Example: frequency = 4, n = 1

   [('museum', 'innocence')]

   frequency = 2, n = 3

   [('innocence', 'pamuk'), ('fall', 'love'), ('words', 'stars')]

   frequency = 1, n = 5

   [('pamuk', 'created'), ('created', 'actual'), ('actual', 'museum'), ('innocence', 'consisting'), ('consisting', 'everyday')]

5. Collocations help us find out that which pairs of words are more probable to occur. You are expected to write a function, which takes bigrams as parameters, and returns the top 10 bigrams.

   Example:

   [('men', 'fall'), ('museum', 'innocence'), ('words', 'stars'), ('admired', 'design'), ('also', 'showed'), ('april', 'mart'), ('arte', 'moderna'), ('beauties', 'seem'), ('collaborated', 'documentary'), ('collaboration', 'culminated')]

6. Create a function that returns the score information of the bigrams that are equal to or more frequent than 2.

   Example:

   [(('museum', 'innocence'), 1.9143835616438356), (('men', 'fall'), 1.6964607224818458), (('words', 'stars'), 1.3851543795846066), (('fall', 'love'), 1.3415656054018743), (('innocence', 'pamuk'), 1.24470166277358)]

7. You are expected to create a function that produces a list of words. Each word will have speech tag along with them. You need to make use of POS-taggers.

   Example:

   [('Pamuk', 'NNP'), ('created', 'VBD'), ('an', 'DT'), ('actual', 'JJ'), ('Museum', 'NN'), ('of', 'IN'), ('Innocence', 'NNP'), (',' , ','), ('consisting', 'VBG'), ('of', 'IN'), ('everyday', 'JJ'), ('objects', 'NNS'), ('tied', 'VBN'), ('to', 'TO'), ('the', 'DT') , ('narrative', 'JJ'), (',', ','), ('and', 'CC'), ('housed', 'VBD'), ('them', 'PRP'), ('at', 'IN'), ('an', 'DT'), ('Istanbul', 'N NP'), ('house', 'NN'), ('he', 'PRP'), ('purchased', 'VBD'), ('.', '.'), ('Pamuk', 'NNP'), ('collaborated', 'VBD'), ('on', 'IN'), ('a', 'DT'), ('documentary', 'JJ'), ('“', '“'), ('The', 'DT'), ('Innocence', 'NNP'), ('Of', 'IN'), ('Memories', 'NNPS'), ("”", "”"), ('[', 'VBZ'), ('17', 'CD'), (']', 'NN'), ('[', 'VBD'), ('18', 'CD'), (']', 'NN'), ('that', 'WDT'), ('expanded', 'VBD'), ('on', 'IN'), ('his', 'PRP$'),...]

8. Write a function **numOfTags** that takes tagged list and returns only the most common tags.

   Example:

   [('NNP', 54), ('IN', 41), ('NN', 34), ('DT', 22), (',', 18), ('JJ', 14), ('NNS', 14), ('VBD', 13), ('CD', 13), ('.', 12)]

9. Write a function, which takes two parameters (one of them for tagged text, and the other one is for a tag), displays the words in descending order for the specified tag.

   Example:

   Here is the list of all the noun types of the text:

   ['Pamuk', 'Museum', 'Innocence', 'objects', 'Istanbul', 'house', 'Memories', 'Museum', 'Dreams', 'version', 'love', 'story', 'Grant', 'Gee', '', 'film', 'Snow', 'love-stories', 'men', 'women', 'sight', 'portrayals', 'reasons', 'intensity', 'way', 'stories', 'heroes', 'beauties', 'decrepit', 'loneliness', 'Grazia', 'Toderi', 'work', 'collaboration', 'exhibition', 'Words', 'Stars', 'April', 'MART', 'Museo', 'Arte', 'Moderna', 'Contemporanea', 'Trento', 'Rovereto', 'inclination', 'man', 'space', 'vocation', 'stars', 'show', 'Gianfranco', 'Maraniello', 'November', 'March', 'Palazzo', 'Madama', 'Piazza', 'Castello', 'Turin', 'Infini-to', 'Planetarium', 'Infini.to', 'Planetario', 'di', 'Torino', "dell'Astronomia", 'e', 'dello', 'Spazio', 'invitation']

10. You are expected to list all the words with number of occurrences information, frequency information and rank information along with their speech tags. After having acquired this set of information, you need to make an observation that whether your text corpus conforms the claim that are stated by Zipf's Law or not (e.g. is the text balanced? , which of the tags or words are frequent or rare, what is the importance of the most frequent word? etc.).