Date : 02/04/2020

Assignment No : 02

Machine Learning (CS-603)

Name : Pulkit Mishra

Reg No: 258

Roll Number : 60

Problem Statement

Given:

94 Bengali literature documents in WX format where each document contains data in the following format:

1. The sentences (a group of words) are separated by a blank lines i.e. each blank line in the document specifies the starting of a new sentence.
2. Each non-blank line contains a word which is already POS tagged.

Task:

1. Apply machine learning tool and algorithm to find out the followings in the set of 94 documents:

- The meaningful topics covered by the documents.
- The keywords for each of those topics.

2. Submit a report (softcopy) mentioning how you have performed this task i.e. the algorithm, code and output in detailed format.

Algorithm

A recurring subject in NLP is to understand a large corpus of texts through topic extraction.

LDA (*short for Latent Dirichlet Allocation*) is an unsupervised machine-learning model that takes documents as input and finds topics as output. The model also says in what percentage each document talks about each topic. Each topic is represented as a weighted list of words

Advantages

1. Fast
2. Intuitive : Modelling topics as weighted lists of words is a simple approximation yet a very intuitive approach if you need to interpret it. No embedding nor hidden dimensions, just bags of words with weights.
3. Can predict topics for new unseen documents : If the new documents have the same structure and should have more or less the same topics, it will work.

Main disadvantages of LDA

1. Lots of fine-tuning
2. It needs human interpretation : Topics are found by a machine. A human needs to label them in order to present the results to non-experts.

Generally text processing pipeline for LDA looks something like this

1. **Removing Encodings**
    - All of the special hidden characters, escape sequences, control characters need to be removed.
    - All such characters were already removed in the given dataset
2. **Remove Punctuations**
    - All of the punctuations need to be removed.
    - All such punctuations were already removed in the given dataset
3. **Part of Speech Tag**
    - We are usually not interested in all words when doing LDA. For example, when doing Topic Modeling as in this case, it is advisable to stick to Nouns or Adjective/Noun pairs. We certainly don’t need articles or pronouns while building the LDA model. So by doing POS tagging, we can extract the parts we actually care about.
    - The given dataset was already POS tagged with the first word of each line being the token and the last word being the tag.
    - I prepared three different datasets out of the dataset
        a. First dataset (data_A.txt) has all the words

    b. Second dataset (data_N.txt) only has the nouns
    c. Third dataset (data_NandV.txt) has the nouns and verbs
- This is done in get_tok() method of data_prep.py by
    a. Iterating through each line of each page
    b. Selecting the first and last words of each line
    c. All first words are added to the first datatset
    d. First words whose corresponding last word begins with N are added to the second dataset
    e. First words whose corresponding last word begins with N or V are added to the third dataset

4. **Tokenizing Sentences and Words**
    - Input to the LDA model is supposed to be a list of lists
    - This is done in get_all() method of data_prep.py by
        a. Iterating through all the documents
        b. get_tok() method is called for each document which returns list of words in the document
        c. Each such list is appended to get a list of lists
        d. Data is dumped in json format
5. **Remove stop words**
    - A lot of common words can be present in document which do not play a role in topic modeling and can add noise tour LDA model
    - Since I do not know Bengali I could not create any custom stop word list o I have not performed stop word removal
6. **Lemmatize or Stemmins**
    - There is a need to remove redundant words that are present due to either multiple conjugations or plurality. Very useful in removing the dimension.
    - Since I do not know Bengali I have not done any such cleaning of the data
7. **LDA implementation**
    - read_data method of lda.py reads data from the file and converts it into a list
    - build_model method of lda.py file builds the lda model and returns it
    - To implement the LDA in Python, I have used the *gensim* package
    - The parameters given to the model are as follows
        a. the number of topics is equal to **num_topics :** In the experiments made I tried several values and am submitting output of 3 and 9 for dataset 2 and 3 and output of 4 and 8 for dataset 1
        b. the *[distribution of the]* number of words per topic is handled by **eta**
        c. the *[distribution of the]* number of topics per document is handled by **alpha**
    - Results method in lda.py writes the topics found to a file
8. **Data Visualization**
    a. *pyLDAvis* package is *used* to visualize the LDA model

Code

1. data_prep.py

```
# Below code is for data preparation

import os
import json

def get_tok(file):
    f = open(file, "r")
    list = []
    for line in f:
    words = line.split()
    if words and (words[-1][0] == "N" or words[-1][0] == "V"):
    list.append(words[0])
    return list

def get_all():
    mega = []
    basepath = 'testdata/new/'
    for entry in os.listdir(basepath):
    if os.path.isfile(os.path.join(basepath, entry)):
    mega.append(get_tok(os.path.join(basepath, entry)))
    with open('data_NandV.txt', 'w') as outfile:
    json.dump({"name": mega},outfile)

get_all()
```

2. lda.py

```
# Below code is for implementing LDA, getting output and data visulaization

from gensim import corpora, models
import numpy as np
import json

def read_data():
    with open('data_NandV.txt') as json_file:
    data = json.load(json_file)
    return data

def build_model(data):
    from gensim import corpora, models
    print(type(data['name']))
    print(type(data['name'][0]))
    list_of_list_of_tokens = data['name']
    dictionary_LDA = corpora.Dictionary(list_of_list_of_tokens)
    dictionary_LDA.filter_extremes(no_below=3)
    corpus = [dictionary_LDA.doc2bow(list_of_tokens) for list_of_tokens in
list_of_list_of_tokens]

    num_topics = 9
    lda_model = models.LdaModel(corpus, num_topics=num_topics, \
            id2word=dictionary_LDA, \
            passes=10, alpha=[0.001]*num_topics, \
            eta=[0.001]*len(dictionary_LDA.keys()))
    return lda_model

def results(lda_model):
    ans = []
    for i,topic in lda_model.show_topics(formatted=True, num_topics=9,
num_words=10):
    ans.append(str(i)+": "+ topic)
    ans.append("\n")
    outf = open("LDA_NandV_9topics.txt",'w')
    outf.writelines(ans)
    outf.close()

def visulaize(lda_model):
    import pyLDAvis
    import pyLDAvis.gensim
    vis = pyLDAvis.gensim.prepare(topic_model=lda_model, corpus=corpus,
dictionary=dictionary_LDA)
    pyLDAvis.save_html(vis, "LDA_NandV_9topics.html")

data = read_data()
lda_model = build_model(data)
results(lda_model)
visulaize(lda_model)
```

Output

Total 6 outputs have been generated as the given dataset was split into three datasets as specified above.

- For dataset 1, number of topics were kept 4 in one experiment and 8 in the other
    - For number of topics = 4, data visualization can be found in A/LDA_A_4topics.html
        a. 0.016*"rameSa" + 0.014*"ramA" + 0.008*"sureSa" + 0.006*"mahima" + 0.006*"dAkwAra" + 0.005*"jyATAimA" + 0.005*"rameSera" + 0.004*"wAxera" + 0.004*"BArawI" + 0.003*"ramAra"

        b. 0.016*"rameSa" + 0.009*"kexArabAbu" + 0.008*"ramA" + 0.007*"beNI" + 0.005*"rameSera" + 0.004*"gobinxa" + 0.004*"jyATAimA" + 0.004*"sureSa" + 0.003*"BEraba" + 0.003*"ramAra"

        c. 0.028*"BArawI" + 0.020*"apUrba" + 0.015*"dAkwAra" + 0.006*"xAxA" + 0.005*"apUrbara" + 0.004*"weoyZArI" + 0.004*"BArawIra" + 0.004*"xeSera" + 0.004*"sumiwrA" + 0.003*"SaSI"

        d. 0.027*"acalA" + 0.020*"sureSa" + 0.013*"mahima" + 0.010*"acalAra" + 0.007*"sureSera" + 0.005*"mqNAla" + 0.004*"mahimera" + 0.003*"kexArabAbu" + 0.003*"bqxXa" + 0.003*"gAdZi"
    - For number of topics = 8, data visualization can be found in A/LDA_A_8topics.html
        a. 0.045*"apUrba" + 0.021*"weoyZArI" + 0.020*"BArawI" + 0.008*"apUrbara" + 0.007*"weoyZArIra" + 0.006*"tAkA" + 0.004*"nIce" + 0.003*"BArawIra" + 0.003*"banXa" + 0.003*"bAsAyZa"

        b. 0.032*"rameSa" + 0.020*"ramA" + 0.010*"beNI" + 0.009*"rameSera" + 0.009*"jyATAimA" + 0.006*"ramAra" + 0.006*"gobinxa" + 0.005*"biSbeSbarI" + 0.005*"BEraba" + 0.004*"wora"

        c. 0.025*"mqNAla" + 0.009*"BAi" + 0.008*"apUrba" + 0.008*"xixi" + 0.007*"yawIna" + 0.006*"sejaxi" + 0.006*"Celera" + 0.006*"mAyZera" + 0.005*"CedZe" + 0.005*"apUrbara"

        d. 0.023*"acalA" + 0.022*"sureSa" + 0.012*"mahima" + 0.010*"acalAra" + 0.007*"kexArabAbu" + 0.007*"sureSera" + 0.004*"mahimera" + 0.003*"bqxXa" + 0.002*"sureSabAbu" + 0.002*"lAgilena"

e. 0.029*"apUrba" + 0.016*"BArawI" + 0.012*"sumiwrA" + 0.011*"xeSera" + 0.006*"apUrbabAbu" + 0.005*"apUrbara" + 0.005*"mAnuRera" + 0.005*"oi" + 0.004*"lokati" + 0.004*"ApanAxera"

f. 0.038*"mahima" + 0.023*"sureSa" + 0.012*"acalA" + 0.005*"pArabe" + 0.004*"seo" + 0.004*"niwe" + 0.004*"wAxera" + 0.004*"kara" + 0.004*"xaroyZAna" + 0.004*"sbAmIra"

g. 0.017*"acalA" + 0.009*"mqNAla" + 0.008*"sureSa" + 0.006*"kexArabAbu" + 0.005*"apUrba" + 0.004*"gAdZi" + 0.003*"mahima" + 0.003*"sureSera" + 0.003*"acalAra" + 0.003*"mqNAlera"

h. 0.033*"BArawI" + 0.021*"dAkwAra" + 0.011*"apUrba" + 0.008*"xAxA" + 0.005*"sumiwrA" + 0.005*"SaSI" + 0.005*"BArawIra" + 0.004*"wAxera" + 0.004*"xeSera" + 0.003*"apUrbara"

- For dataset 2, number of topics were kept 3 in one experiment and 9 in the other
  - For number of topics = 3, data visualization can be found in A/LDA_N_3topics.html

    a. 0.033*"sureSa" + 0.015*"kexArabAbu" + 0.013*"mahima" + 0.012*"sureSera" + 0.008*"mqNAla" + 0.008*"mahimera" + 0.006*"acalAra" + 0.005*"acalA" + 0.005*"gAdZi" + 0.004*"sbAmIra"

    b. 0.046*"BArawI" + 0.028*"dAkwAra" + 0.011*"xAxA" + 0.008*"BArawIra" + 0.008*"sumiwrA" + 0.007*"xeSera" + 0.006*"weoyZArI" + 0.005*"apUrbabAbu" + 0.005*"SaSI" + 0.004*"apUrbara"

    c. 0.040*"rameSa" + 0.026*"ramA" + 0.012*"rameSera" + 0.012*"jyATAimA" + 0.010*"beNI" + 0.008*"ramAra" + 0.006*"BEraba" + 0.005*"gobinxa" + 0.005*"rAmabAbu" + 0.005*"beNIra"

  - For number of topics = 9, data visualization can be found in A/LDA_N_9topics.html

    a. 0.051*"rameSa" + 0.029*"ramA" + 0.015*"jyATAimA" + 0.015*"rameSera" + 0.011*"beNI" + 0.010*"ramAra" + 0.007*"BEraba" + 0.007*"gobinxa" + 0.006*"biSbeSbarI" + 0.006*"beNIra"

    b. 0.032*"BArawI" + 0.007*"Celera" + 0.007*"tAkA" + 0.007*"maxa" + 0.007*"mAyZera" + 0.006*"apUrbara" + 0.006*"rAga" + 0.006*"saMsAre" + 0.006*"mAke" + 0.006*"Xarma"

c. 0.029*"ramA" + 0.019*"beNI" + 0.011*"jyATAmaSAi" + 0.011*"suramA" + 0.010*"ebAra" + 0.009*"tAkA" + 0.009*"snAna" + 0.009*"rameSera" + 0.009*"rAmabAbu" + 0.008*"anekakRaNa"

d. 0.043*"sureSa" + 0.019*"kexArabAbu" + 0.016*"mahima" + 0.015*"sureSera" + 0.009*"mahimera" + 0.007*"acalAra" + 0.007*"mqNAla" + 0.006*"acalA" + 0.005*"gAdZi" + 0.004*"sureSabAbu"

e. 0.040*"dAkwAra" + 0.017*"BArawI" + 0.011*"sumiwrA" + 0.010*"apUrbabAbu" + 0.010*"dAkwArabAbu" + 0.009*"rAmaxAsa" + 0.006*"BAi" + 0.006*"Xarma" + 0.005*"apUrbara" + 0.004*"aXikAra"

f. 0.035*"mqNAla" + 0.031*"BAi" + 0.016*"xixi" + 0.010*"Age" + 0.006*"rAmaxAsa" + 0.005*"acalAra" + 0.005*"xeSera" + 0.005*"mqNAlera" + 0.005*"kAne" + 0.005*"sebA"

g. 0.029*"weoyZArI" + 0.013*"sAheba" + 0.010*"weoyZArIra" + 0.007*"nIce" + 0.007*"xeSera" + 0.006*"rAmaxAsa" + 0.006*"rAga" + 0.006*"APisera" + 0.005*"puliSera" + 0.005*"bAbu"

h. 0.046*"BArawI" + 0.020*"dAkwAra" + 0.007*"xAxA" + 0.007*"BArawIra" + 0.006*"xeSera" + 0.005*"sumiwrA" + 0.004*"apUrbabAbu" + 0.004*"tAkA" + 0.003*"mAnuRera" + 0.003*"gAdZi"

i. 0.055*"dAkwAra" + 0.048*"BArawI" + 0.027*"xAxA" + 0.019*"SaSI" + 0.014*"sumiwrA" + 0.014*"BArawIra" + 0.009*"kabi" + 0.006*"mAnuRera" + 0.005*"xeSera" + 0.005*"saMsAre"

- For dataset 3, number of topics were kept 3 in one experiment and 9 in the other
    - For number of topics = 3, data visualization can be found in A/LDA_NandV_3topics.html

a. 0.020*"sureSa" + 0.013*"ramA" + 0.011*"rameSa" + 0.009*"mahima" + 0.006*"sureSera" + 0.004*"mahimera" + 0.004*"beNI" + 0.004*"ramAra" + 0.003*"rameSera" + 0.003*"acalAra"

b. 0.023*"BArawI" + 0.015*"dAkwAra" + 0.006*"xAxA" + 0.005*"sureSa" + 0.005*"kexArabAbu" + 0.004*"BArawIra" + 0.004*"sumiwrA" + 0.004*"xeSera" + 0.003*"apUrbabAbu" + 0.003*"SaSI"

c. 0.017*"rameSa" + 0.007*"weoyZArI" + 0.006*"jyATAimA" + 0.005*"mqNAla" + 0.005*"BAi" + 0.005*"rameSera" + 0.004*"BEraba" + 0.004*"gobinxa" + 0.003*"xixi" + 0.003*"tAkA"

- For number of topics = 9, data visualization can be found in A/LDA_NandV_9topics.html

    a. 0.014*"sureSa" + 0.010*"kexArabAbu" + 0.005*"sureSera" + 0.005*"mahima" + 0.004*"mahimera" + 0.003*"mqNAla" + 0.003*"weoyZArI" + 0.003*"gAdZi" + 0.003*"sbAmIra" + 0.003*"nIce"

    b. 0.037*"BArawI" + 0.009*"rAmaxAsa" + 0.008*"weoyZArI" + 0.004*"apUrbara" + 0.004*"haiyZACila" + 0.004*"apUrbabAbu" + 0.004*"rAga" + 0.004*"CedZe" + 0.004*"mAke" + 0.004*"apUrba"

    c. 0.041*"rameSa" + 0.026*"ramA" + 0.013*"rameSera" + 0.012*"jyATAimA" + 0.010*"beNI" + 0.009*"ramAra" + 0.006*"BEraba" + 0.006*"gobinxa" + 0.005*"beNIra" + 0.004*"ne"

    d. 0.045*"mqNAla" + 0.018*"kexArabAbu" + 0.007*"kRamA" + 0.007*"hAlaxAra" + 0.006*"lAgilena" + 0.005*"mqNAlera" + 0.005*"WAkabe" + 0.005*"ewakAla" + 0.005*"uTilena" + 0.005*"meyZe"

    e. 0.022*"bqxXa" + 0.021*"GqNA" + 0.020*"hauka" + 0.015*"bApera" + 0.011*"sAkRI" + 0.011*"suramA" + 0.011*"kAke" + 0.011*"hacce" + 0.011*"parei" + 0.010*"pArawena"

    f. 0.015*"sureSa" + 0.009*"BAi" + 0.005*"xixi" + 0.005*"sureSera" + 0.005*"mahima" + 0.004*"kexArabAbu" + 0.003*"acalAra" + 0.003*"kAne" + 0.003*"haibe" + 0.003*"tAkA"

    g. 0.019*"sureSa" + 0.008*"mahima" + 0.007*"sureSera" + 0.006*"rAmabAbu" + 0.004*"weoyZArI" + 0.003*"acalAra" + 0.003*"acalA" + 0.003*"sureSabAbu" + 0.003*"gAdZi" + 0.003*"jyATAmaSAi"

    h. 0.040*"BArawI" + 0.030*"dAkwAra" + 0.012*"xAxA" + 0.008*"sumiwrA" + 0.008*"BArawIra" + 0.006*"xeSera" + 0.005*"SaSI" + 0.005*"apUrbabAbu" + 0.004*"mAnuRera" + 0.003*"dAkwArera"

    i. 0.042*"sureSa" + 0.018*"mahima" + 0.015*"sureSera" + 0.012*"mahimera" + 0.010*"banXu" + 0.008*"gAdZi" + 0.006*"nIce" + 0.006*"acalAra" + 0.006*"bAbAra" + 0.006*"xoRa"