**ASSIGNMENT 8**

**Topic Modeling**

**Introduction:**

Vast amount of unstructured text data is available online and offline consisting of corpus of documents. These documents can be comprised of multiple topics within them. To be able to identify various topics and classify this huge corpus of documents into a bucket of documents based on the topics we use Topic Modeling. Topic Modeling allows you to a find themes in a collection of documents and to get a bird’s eye view of the document. There are several algorithms used for Topic Modeling. One of the useful and widely used algorithm is LDA (Latent Dirichlit Allocation). LDA assumes that a document can be considered as a mixture of different topics and each topic is a distribution over a vocabulary of words. The words occur in a different topic with different weightage. Some words are more frequent in some topics and not in every topic. Every topic will have a list of keywords which can be used to determine the topic of the text. LDA treats each topic assignment separate and doesn’t correlate them to create 1 topic.

This task is about Topic Modeling using LDA algorithm to summarize the topics in the floor debate 110th Congress (House Only) collection of text using Mallet Software. The goal of this assignment is to find out if we can list out the main topics correctly being discussed in this text collection along with accurately predicting number of topics in the corpus. The topics can be inferred by looking at the key words generated by this algorithm and to see how accurately these words come up for anyone to infer the topics in text.

**Data Definition:**

The Dataset consists of 4 subfolders with ‘m’ meaning ‘Male’ and ‘f’ meaning ‘Female’ while ‘r’ is for ‘Republican’ and d is for ‘Democrat’. Each subfolder has number of text documents which have number of topics in them. There are 50 text documents in 110-f-d folder, 18 in 110-f-r, 202 in 110-m-d, 159 in 110-m-r with a total of 429 documents in the whole text collection. There are only 15% of documents by females as compared to males which are 85%. The set is not balanced on that aspect and would be more centric towards male opinion in the congress house. All these documents are combined and placed in the mallet folder under sample-data folder to be analyzed for topic modeling.

**Methods:**

The text collection to be used for topic modeling is placed inside sample-data folder in Mallet folder. The text can be analyzed on individual basis or as a whole collection. First, we convert all the documents to Mallet format for it to be analyzed. The input is the collection of these documents with several options for pre-processing such as tokenization, using unigram, bigrams or trigrams, removing stopwords, converting all the words to lowercases. These parameters can be tuned to get what is required as an output for topic modeling. The output using these combinations of parameters is a mallet file we specify with .mallet extension. This .mallet file is used for topic modeling where we can specify number of topics for the documents to be classified in and tune these topics to get the best number of topics. The output of modeling consists of 3 files. These files include .gz file standard output file, keywords.txt file which consists of all the keywords based on the number of topics we specified initially, and lastly topics.txt which consists of probability that a document belongs to that topic. Each document will have probabilities for each topic and selecting the largest probability to classify document belongs to this topic.

The number of topics parameter while building model is changed to get highest value of LL/tokens at the end of output of mallet to select best number of topics. Also, usage of bigrams while inputting the data improved the performance. These parameters were tuned to get the best possible topics and count of topics.

**Results:**

During the fist step of Data Preparation, the input text collection is vectorized, stop words are removed. For this process, various parameters used are - -keep-sequence which keeps the document as a sequence of word features to keep the context information. The other parameters tried to get best results are - -remove-stopwords which removes the stopwords listed in mallet in stoplists/en file.

Starting with the removing default stop words using only unigrams along with number of topics 10 the words did not give clear idea of what could the topics be specifically. Also, the LL/tokens value for this combination was -9.12197. The number of topics were limited to Iraq sanctions, Energy and oil issues, Housing and 2 3 topics had the same kind of keywords like Iraq, Bill, Tax, Housing which pointed to same topic and not different topics. 10 topics with their keywords gave less than 10 topics by looking at their keywords. Also, I found that there were several words such as representatives, government, American, Speaker as this is dialogue between the government officials in the house so these words are most common and filtering these words would be beneficial to get new words in keywords to better understand the topics. For this, I used - -extra-stopwords parameter to specify my list of stop words like government, house, speaker, representatives in a file named extra-stopwords.txt placed in stoplists folder. This is in addition to default English stop words being filtered out. Using bigrams gave more meaningful words mass\_destruction, affordable\_housing etc as a bigram more meaningful than using unigrams. Also, the - -preserve-case option was used to not convert the letters in lowercase as this helps in running it faster than converting it into lowercase which takes a bit more time of approximately 1 to 2 minutes. These options/parameters were tuned to get the best processed output for the mallet model building through topic training in the next step. The table compares use of unigrams versus use of bigrams. The table shows that for same number of topics bigrams are better. Also, the words generated more meaningful and allow better understanding than unigrams alone.

|  |  |  |
| --- | --- | --- |
|  | Unigram (Number of topics 40) | Bigram (Number of topics 40) |
| LL/tokens value | -9.33286 | -14.20727 |
| Run Time | 19 mins 46 secs | 21 mins 58 secs |

The use of extra stop words such as government, representatives etc. based on the context improved the run time, gave better key words and helped to get higher LL/tokens value keeping the same number of topics to be 40. The table below shows the comparison of usage of both and their effects.

|  |  |  |
| --- | --- | --- |
|  | Default Stop words removal | Extra stop words removal |
| LL/tokens value | -9.09197 | -9.33286 |
| Run time | 20 mins 10 sec | 19 mins 46 secs |

The .mallet file at the output of Data Pre-Processing was used to train the model. To train the topics build, using the parameter - -optimize-interval to 20 which enables hyperparameter tuning and facilitates the prominence of some topics over others. This changes the Prominence values which appear next to Topic ID in the keywords.txt file. These higher and lower prominence values make the words more valuable in one topic and less valuable in other topics. This also helps to get better topics as the weights are changed to reflect the weightage of different words in them. Using optimize-interval 10 gave LL/tokens score to be -14.17596 and using optimize-interval 20 gave LL/tokens score to be -14.20727 a slight but better LL/token score so I chose to go with optimize-interval 20. The - -num-topics are varied to get the highest LL/tokens score to make the best possible guess of the number of topics. The highest the LL/tokens (Log Likelihood) value, the better is the tuning of the topics. Highest Log Likelihood gets the best number of topics to be used for topic modeling. Sometimes this highest value can give meaningless keywords, so this has to be taken care of. Not to use too many numbers of topics to get higher LL/token value and at the same time get meaningless words. The number of topics for this combination was tried for 40, 45 and 50 topics. LL/token values, run time duration, number of tokens used were recorded to get the best possible values of number of topics. The below table depicts the variation of num-topics parameter.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Num-topics 40 | Num-topics 45 | Num-topics 50 |
| LL/tokens values | -14.20727 | -14.18852 | -14.15432 |
| Run time | 25 mins 58 secs | 35 mins 47 sec | 43 mins 50 secs |
| Tokens | 7338823 | 7338569 | 7338778 |

The above table shows the comparison of using different topic values to tune the parameter. From this table, the number of topics with 40 is chosen as it gives the best LL/tokens along with running for almost 10 mins less as compared to 45 or 50 topics. The words that can be found out of the keywords.txt file to see if selecting the number of topics 40 does create good meaningful words came to be good. The words like occupation\_Iraq, military\_contractors tell us that the topic being discussed is related to Iraq occupation and the military. The Iraqi people’s occupation of being in military in a contract. The use of bigrams did help in giving better results in terms of both keywords selection as well as number of topics tuning. Similarly, fossil\_fuels years\_ago, chart\_shows, peak\_oil, million\_barrels kind of words talk about the topic of fossil fuels years ago and the barrels at that peak time in USA. The file topics.txt gives the probability of document belonging to a particular topic based on the highest probability. For example, 110\_abercrombie\_x\_hi.txt file had the best probability of being classified into topic 14 with 0.44 probability. After going to check the words in the topic 14 the words such as health\_centers, mental\_health were mentioned which related to this document as the document talks about Pearl Harbor and other incidents about the troops with mental\_illness, bills to build health\_centres which makes sense if we were to classify this document based on the topic we found in keywords. So, these files at the output of the mallet topic training help to summarize the keywords to identify important topics from them and topics file can be used to see the probability that the document is classified into that topic for topic modeling.

The various topics that can be inferred from the keywords file are Medicines and Drugs, Money of people, Education and Higher Education along with Education Loans, Veterans World War, Taxpayers and Highest/Lowest Taxpaying states along with any tax relief, Trade agreements and employment, Judiciary Civil Rights, Oil and Gas prices along with their rate changes, Wall street, Nuclear Energy and weapons, Health Insurance and children benefits, Infrastructure including Transportation passenger rails, Intelligence committees and authorities, National Defense and Foreign Intelligence, Science & Technology along with Math Science, Troops and Military, Global Warming, Affordable Housing, Iraq situation and the troops there, Fiscal Year Financial markets, Labor Markets and their issues are some the topics that can be inferred from the keywords. It includes the major topics in discussion like Defense, Health, Education, Gas and Oil Prices, Judiciary, Economy.

**Conclusion:**

Topic Modeling using Mallet was very helpful to find the topics inside a large corpus of texts of approximately 400 in number. The use of bigrams is better for both number of topics as well as the keywords generated to identify topics. Usage of extra stop words based on the context can give better results by giving accurate keywords instead of these keywords being repeated. This also helped in making inference about the topic better. Number of topics 40 tuned gave the highest LL/tokens along with keeping the meaningfulness of the words. More number of topics did not lead to higher LL/tokens score along with running for almost 10 mins longer, making 40 the best suited number of topics. The use of optimize-interval parameter allows to give different weights called Prominence values to each topic and helps in deciding better topics based on higher and lower values given to topic for different keywords to belong to one topic more closely than to the other topics. The probability of document belonging to a certain topic we modelled is helpful to put the documents of same topic in a same bucket, as if creating a self of documents of similar topics for easy and concise understanding of the document. Some of the techniques such as wordcloud creation from the keywords which were not used in this assignment can be very helpful. Also, using this trained topic output we can predict the topics of unseen documents. This can help in testing the effectiveness of the algorithm for more generalized models. Topic Modeling using LDA algorithm gives a very effective bird’s eye overview of the documents in hand by summarizing in a nutshell which main themes are being talked about. In practical, unlabeled text documents are available in more abundance as compared to labeled documents so Topic Modeling is very useful to see the main ideas of documents without going through the whole of the article.