NLP Project on LDA Topic Modelling Python using RACE Dataset

Using the RACE dataset to extract a dominant topic from each document and perform LDA topic modeling in python.

Project Description

Business Context

With the advent of big data and Machine Learning along with Natural Language Processing, it has become the need of an hour to extract a certain topic or a collection of topics that the document is about. Think when you analyze or go through thousands of documents and categorize under 10 – 15 buckets. How tedious and boring will it be?

Thanks to Topic Modeling, instead of manually going through numerous documents, with the help of Natural Language Processing and Text Mining, each document can be categorized under a certain topic.

Thus, we expect that logically related words will co-exist in the same document more frequently than words from different topics. For example, in a document about space, it is more possible to find words such as: planet, satellite, universe, galaxy, and asteroid. Whereas, in a document about wildlife, it is more likely to find words such as: ecosystem, species, animal, and plant, landscape. A topic contains a cluster of words that frequently occur together. Topic modeling can connect words with similar meanings and distinguish between uses of words with multiple meanings.

A sentence or a document is made up of numerous topics and each topic is made up of numerous words.

Data Overview

The dataset has odd 25000 documents where words are of various nature such as Noun, Adjective, Verb, Preposition and many more. Even the length of documents varies vastly from having a minimum number of words in the range around 40 to maximum number of words in the range around 500. Complete data is split 90% in the training and the rest 10% to get an idea how to predict a topic on unseen documents.

Objective

To extract or identify a dominant topic from each document and perform topic modeling.

Tools and Libraries

We will be using Python as a tool to perform all kinds of operations.

Main Libraries used are

- Pandas for data manipulation, aggregation
- Matplotlib and bokeh for visualization of how documents are structured.
- NumPy for computationally efficient operations.
- Scikit Learn and Gensim packages for topic modeling
- nltk for text cleaning and preprocessing
- TSNE and pyLDAvis for visualization of topics

Approach

Topic EDA

- Top Words within topics using Word Cloud
- Topics distribution using t-SNE
- Topics distribution and words importance within topics using interactive tool pyLDAvis

Documents Pre-processing

- Lowering all the words in documents and removing everything except alphabets.
- Tokenizing each sentence and lemmatizing each word and storing in a list only if it is not a stop word and length of a word is greater than 3 alphabets.
- Join the list to make a document and keep the lemmatized tokens for NMF Topic Modelling.
- Transforming the above pre-processed documents using TF IDF and Count Vectorizer depending on the chosen algorithm

Topic Modelling algorithms

- Latent Semantic Analysis or Latent Semantic Indexing (LSA)
- Latent Dirichlet Allocation (LDA)
- Non-Negative Matrix Factorization (NMF)
- Popular topic modelling metric score known as Coherence Score
- Predicting a set of topics and the dominant topic for each document
- Running a python script end to end using Command Prompt

Illustration for Each Part

Natural Language Processing (NLP):

Natural language processing (NLP) refers to the branch of computer science—and more specifically, the branch of artificial intelligence or AI—concerned with giving computers the ability to understand text and spoken words in much the same way human beings can.

NLP combines computational linguistics—rule-based modeling of human language—with statistical, machine learning, and deep learning models. Together, these technologies enable computers to process human language in the form of text or voice data and to 'understand' its full meaning, complete with the speaker or writer's intent and sentiment.

NLP drives computer programs that translate text from one language to another, respond to spoken commands, and summarize large volumes of text rapidly—even in real time. There is a good chance you have interacted with NLP in the form of voice-operated GPS systems, digital assistants, speech-to-text dictation software, customer service chatbots, and other consumer conveniences. But NLP also plays a growing role in enterprise solutions that help streamline business operations, increase employee productivity, and simplify mission-critical business processes.

NLP tasks

- Speech recognition
- Part of speech tagging
- Named entity recognition
- Co-reference resolution
- Sentiment analysis
- Natural language generation

Topic Modelling:

Topic modeling is a machine learning technique that automatically analyzes text data to determine cluster words for a set of documents. This is known as 'unsupervised' machine learning because it does not require a predefined list of tags or training data that has been previously classified by humans.

TensorFlow Datasets:

a collection of ready-to-use datasets.

Data Analysis Process:

- **Purpose:** *The Goal of this project is to assign a topic to each document and apply topic modeling. The project will extract 10 topics using the Latent Dirichlet Allocation (LDA) Model. And extracting 25 topics using the Non_Negative Matrix Factorization Model. Then, visualizing topic distributions using t_SNE and pyLDAvis. Finally, for unseen documents topics were predicted using these models.*
- **Scope / Major Project Activities:**

Activity	Description
Read Dataset	read the Tensorflow RACE dataset and convert it into a format that is easier to use.
Data Pre-processing	remove the unnecessary characters or words like stopwords, URLs, Punctuations, etc.
Data Vectorizations	apply TF IDF or countvectorizer on data to be ready to be fed to the model.
LDA Model	Extract 10 topics using LDA models
NMF Model	Extract 25 topics using NMF model and the number what found using coherence score
Visualization	Analyze the topics distribution using WordCloud, t_SNE and pyLDAvis

- **This Project Does not Include:** Any topic outside this list of documents
- **Deliverables:**

Deliverable	Description/ Details
WordCloud	The most frequent words in each topic
t_SNE	topics distribution analysis using t_SNE visualization

pyLDAvis	topics distribution analysis using pyLDAvis visualization

Documents Pre-Processing:

- **Removing punctuations like .,! $( ) * % @**
- **Removing URLs**
- **Removing Stop words:** Stopwords are the commonly used words and are removed from the text as t hey do not add any value to the analysis. These words carry less or no meaning. NLTK library consists of a list of words that are considered stopwords for the English language. Some of them are : [i, me, my, myself, we, our, ours, ourselves, you, you're, you've, you'll, you'd, your, yours, yourself, yourselves, he, most, other, some, such, no, nor, not, only, own, same, so, then, too, very, s, t, can, will, just, don, don't, should, should've, now, d, ll, m, o, re, ve, y, ain, aren't, could, couldn't, didn't, didn't]
- **Lower casing**
- **Tokenization:** In this step, the text is split into smaller units. We can use either sentence tokenization or word tokenization based on our problem statement.
- **Lemmatization:** It stems the word but makes sure that it does not lose its meaning. Lemmatization has a pre-defined dictionary that stores the context of words and checks the word in the dictionary while diminishing.

Corpus:

A corpus is a large and structured set of machine-readable texts that have been produced in a natural communicative setting. Its plural is corpora. They can be derived in different ways like text that was originally electronic, transcripts of spoken language and optical character recognition, etc.

Latent Dirichlet Allocation (LDA):

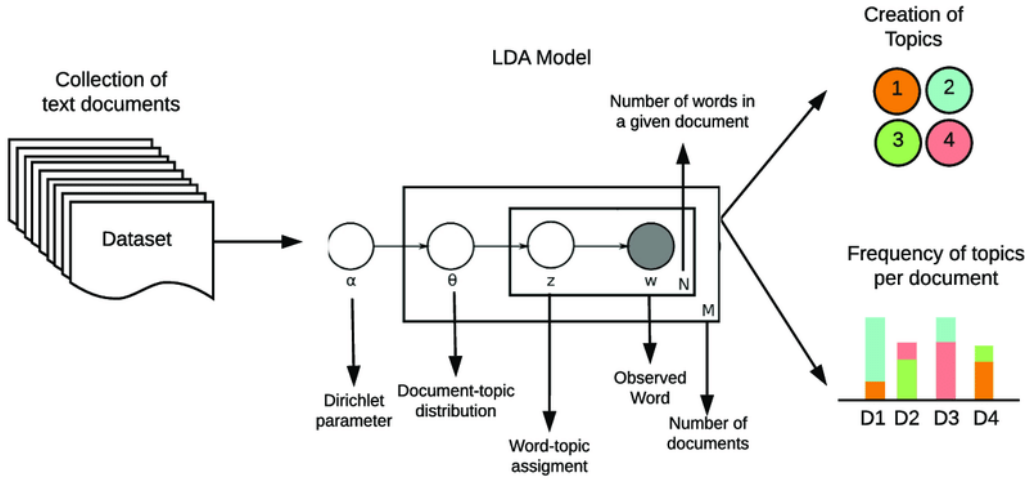
Latent Dirichlet Allocation (LDA) is a popular technique for extracting topics from a corpus. The term latent refers to something that exists but has not yet manifested itself. In other words, latent refers to something that is hidden or concealed.

The topics we want to extract from the data are now "hidden topics" as well. It has not yet been discovered. As a result, the term "latent" is used in LDA. Following the Dirichlet distribution and process comes the Dirichlet allocation.

Latent Dirichlet Allocation (LDA) is a tool and technique for Topic Modeling that classifies or categorises the text into a document and the words per topic using Dirichlet distributions and processes.

The LDA is based on two key assumptions: Documents are made up of topics, and topics are made up of tokens (or words).

These topics generate the words using a probability distribution. The documents are known as the probability density (or distribution) of topics in statistical terms, and the topics are known as the probability density (or distribution) of words.



Non-Negative Matrix Factorization (NMF):

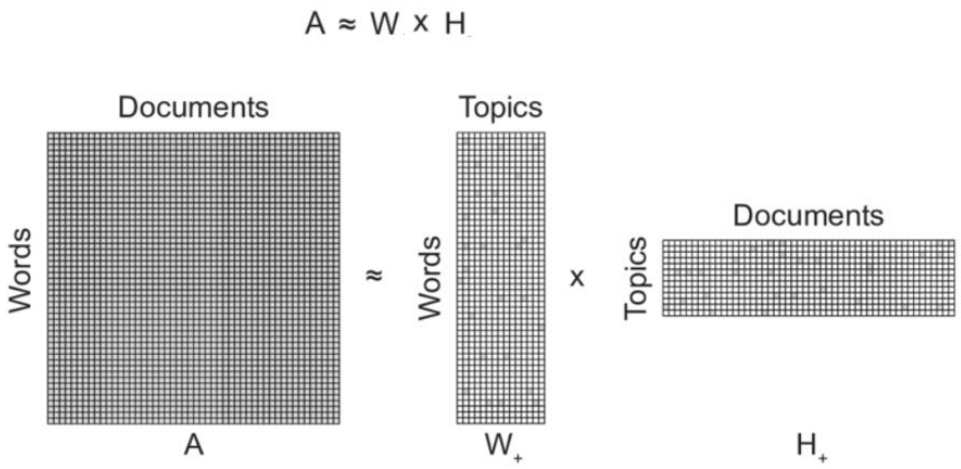
Non-Negative Matrix Factorization is a statistical method for reducing the dimension of input corpus or corpora. Internally, it employs the factor analysis method to assign a lower weightage to words with lower coherence.

1. It is a member of the linear algebra algorithm family, which is used to identify latent or hidden structures in data.
2. It is shown as a non-negative matrix.
3. It can also be used for topic modelling, with the term-document matrix as the input, which is typically TF-IDF normalized.
    - Input: the term-document matrix and the number of topics.
    - Output: Two non-negative matrices of the original n-words by k topics and those same k topics by the m original documents are returned.

    To put it simply, we are using linear algebra for topic modelling.

4. NMF's popularity stems from its ability to automatically extract sparse and easily interpretable factors.

    The following is a visual representation of the above technique:



Coherence Score:

Topic Coherence measures score a single topic by measuring the degree of semantic similarity between high scoring words in the topic. These measurements help distinguish between topics that are semantically interpretable topics and topics that are artifacts of statistical inference.

PyLDAvis:

pyLDAvis is designed to help users interpret topics in a topic model that has been fit to a corpus of text data. The package extracts information from a fitted LDA topic model to inform an interactive web-based visualization.

t-SNE:

t-Distributed Stochastic Neighbor Embedding (t-SNE) is an unsupervised, non-linear technique primarily used for data exploration and visualizing high-dimensional data. In simpler terms, t-SNE gives you a feeling or intuition of how the data is arranged in a high-dimensional space.

The t-SNE algorithm calculates a similarity measure between pairs of instances in the high dimensional space and in the low dimensional space. It then tries to optimize these two similarity measures using a cost function.

References

- ProjectPro.io: https://www.projectpro.io/project-use-case/topic-modelling-python
- IBM Cloud Learn Hub: https://www.ibm.com/cloud/learn/natural-language-processing
- MonkeyLearn Blog: https://monkeylearn.com/blog/introduction-to-topic-modeling/
- TensorFlow: https://www.tensorflow.org/datasets
- AnalyticsVidhya.com: https://www.analyticsvidhya.com/blog/2021/06/text-preprocessing-in-nlp-with-python-codes/
- AnalyticsVidhya.com: https://www.analyticsvidhya.com/blog/2021/06/part-2-topic-modeling-and-latent-dirichlet-allocation-lda-using-gensim-and-sklearn/
- AnalyticsVidhya.com: https://www.analyticsvidhya.com/blog/2021/06/part-15-step-by-step-guide-to-master-nlp-topic-modelling-using-nmf/#:~:text=Non%2DNegative%20Matrix%20Factorization%20is,that%20are%20having%20less%20coherence.
- PyLDAvis: https://pyldavis.readthedocs.io/en/latest/readme.html