Abstract

Reading, understanding and processing text is an enormous aspect in both our daily life and in any project. In today's age of technological advancements, the natural next step is to let machines take over this tedious, yet critical task.

When parsing through a paper, you are likely to skim the text, looking for key ideas that stand out, trying to get a general understanding of what the text is about, and condensing it to information you can retain. Natural Language Processing does the same in theory: It parses through text to add useful numerical structure, resolve ambiguity and draw out key ideas.

Topic Modeling is useful In the specific case where you want to process a large quantity of documents, and draw out information both specific to the document and overarching concepts. It relies on weighting words as a representation of importance and then finding shared instances between the documents.

Topic modeling and machine learning in general are key to eliminating human bias and introducing more objective diversity into our results.

There are several algorithms that do this effectively. A notable few are Non-negative Matrix Factoring (NMF), Latent Dirichlet Allocation (LDA), Principal Component Analysis (PCA), and Latent Semantic Analysis (LSA). In this paper, I will go through the

algorithms, NMF and LDA, and highlight some interesting differences we see in data that is processed through each as a reflection of the algorithm itself.

Data

Two datasets were used in this paper. The first being a collection of 337 Medium articles. Medium is an American open online publishing platform that resides in the heart of the technology, and specifically data science, community. As a result the content mostly revolves around computer science, programming languages, and machine learning.

	author	claps	reading_time	link	title	text
0	Justin Lee	8.3K	11	https://medium.com/swlh/chatbots-were-the-next...	Chatbots were the next big thing: what happene...	Oh, how the headlines blared:\nChatbots were T...
1	Conor Dewey	1.4K	7	https://towardsdatascience.com/python-for-data...	Python for Data Science: 8 Concepts You May Ha...	If you've ever found yourself looking up the s...
2	William Koehrsen	2.8K	11	https://towardsdatascience.com/automated-featu...	Automated Feature Engineering in Python – Towa...	Machine learning is increasingly moving from h...
3	Gant Laborde	1.3K	7	https://medium.freecodecamp.org/machine-learni...	Machine Learning: how to go from Zero to Hero ...	If your understanding of A.I. and Machine Lear...
4	Emmanuel Ameisen	935	11	https://blog.insightdatascience.com/reinforcem...	Reinforcement Learning from scratch – Insight ...	Want to learn about applied Artificial Intelli...

The second dataset used is a collection of 1226258 news headlines from ABC (Australian Broadcasting Corporation) published over the last 18 years. ABC is a popular reputable australian news source. Naturally, we would assume that the news headlines would converge to major headline news in australia as topics

	publish_date	headline_text
0	20030219	aba decides against community broadcasting lic...
1	20030219	act fire witnesses must be aware of defamation
2	20030219	a g calls for infrastructure protection summit
3	20030219	air nz staff in aust strike for pay rise
4	20030219	air nz strike to affect australian travellers

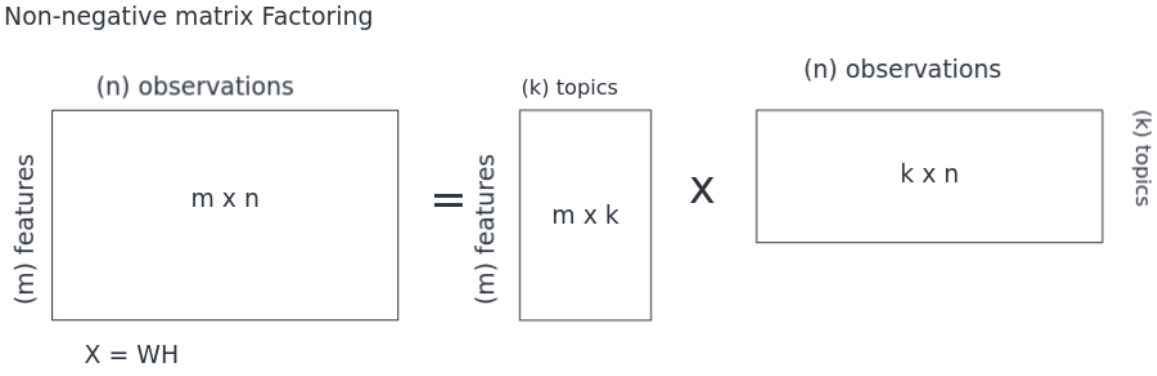
Algorithms

1. NMF

NMF uses the logic of the “distributional hypothesis”, which is the idea that “words that are similar in meaning occur in similar contexts”. This enables us to ignore both semantic and syntactic features of text because we assume word order and word meaning do not matter.

Given a corpus of data, the algorithm will first create a matrix with each document as a column vector and every word in a document will be represented by the row entries. This matrix is then converted into a non-negative matrix, because negativity is insignificant in representing words.

Then we can decompose the matrix such that the 2 matrices are also non-negative as shown below.



In the above diagram X is our non-negative matrix that represents the corpus and all its terms. W and H are it’s factored form.

The number of topics, k, is a hyperparameter, and thus it is up to the programmer to choose a value k that potentially represents the number of topics you expect to see in the corpus. By the nature of matrix multiplication each entry in matrix X should represent the respective dot product of the row vector of W and the column vector of H. Thus each word is composed of a linear combination of all the possible topics for that word and all the possible topics in that document.

This method of factoring accomplishes several things.

(1) Reduce the dimensionality of the system which makes it easier to deal with large datasets
(2) Draw out key topics

In addition, to ensure that the data is trained efficiently and does not include common words. The data is removed of stop words and a weighting scheme is applied to reflect how important each word is to the document.

A popular weighting scheme is tf-idf (term frequency-inverse document frequency), which uses the proportion at which each word appears to add relative importance to each word. The formula is as given below:

$$TF(t_i, d) = \left\{ \frac{f(t_i, d)}{|N_d|} \right\}$$

Ultimately, the goal of NMF is to optimize the cost function ||X- HF|| since it's unlikely that X will factor completely, so we must find the closest approximation.

There are 2 common ways to properly factor the equation: Gradient descent on coordinate matrix or applying the Multiplicative step method, as given below.

$$\mathbf{H}^{n+1}_{[i,j]} \leftarrow \mathbf{H}^{n}_{[i,j]} \frac{((\mathbf{W}^n)^T \mathbf{V})_{[i,j]}}{((\mathbf{W}^n)^T \mathbf{W}^n \mathbf{H}^n)_{[i,j]}}$$

and

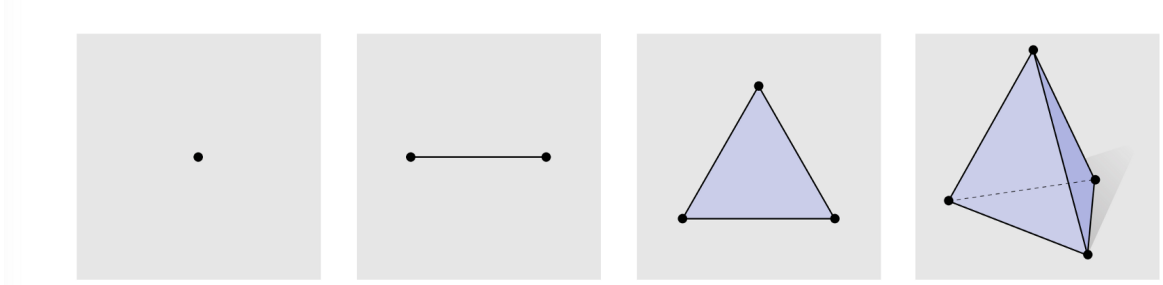
$$\mathbf{W}^{n+1}_{[i,j]} \leftarrow \mathbf{W}^{n}_{[i,j]} \frac{(\mathbf{V}(\mathbf{H}^{n+1})^T)_{[i,j]}}{(\mathbf{W}^n \mathbf{H}^{n+1}(\mathbf{H}^{n+1})^T)_{[i,j]}}$$

2. LDA

LDA functions on a lot of the same assumptions of NMF, like the lack of semantic and syntactic meaning. In addition, it greatly relies on relative knowledge. In essence, LDA assumes in performing it's algorithm that all other word to topic associations are true, and sorts the current word based on its relation to the other words in the corpus.

2.1. Dirichlet Distribution

LDA takes a geometric and probabilistic approach to sorting the same corpus of data. It creates a k-1 dimensional simplex to where each corner represents 1 of the k topics.



It then distributes

Optimizing the hyperparameter k, how greatly choosing a correct k influence the results of the algroithm

Topic modeling is glorified sorting